Quantum Information Processing (2023) 22:47
https://doi.org/10.1007/s11128-022-03805-1

Hybrid quantum-classical scheduling optimization in UAV-enabled IoT networks

Francesco Vista[1,2] · **Giovanni Iacovelli**[1,2] · **Luigi Alfredo Grieco**[1,2]

Received: 27 August 2022 / Accepted: 21 December 2022

Abstract

This work investigates a scenario in which a swarm of unmanned aerial vehicles serves a set of sensor nodes, adopting the time division multiple access scheme. To ensure fair resource allocation and derive an optimal scheduling plan, a combinatorial problem subject to binary constraints is formulated. Thanks to its inherent capabilities, quantum annealing can be used to solve this class of optimization problems. As a result, the original problem is mapped to quadratic unconstrained binary optimization form, in order to be processed by a quantum processing unit. Since state-of-the-art quantum annealers have a limited number of quantum bits (qubits) and limited inter-qubit connectivity, the scheduling plan is obtained by employing a hybrid quantum-classical approach. Then, a comparison with two classical solvers is performed in terms of acquired data, objective function values, and execution time.

Keywords Internet of drones · IoT networks · Quantum computing · Quantum optimization · Quantum annealer

1 Introduction

Quantum computing [1] is a disruptive paradigm that takes advantage of quantum mechanics theory and its properties to solve challenging problems. Nowadays, industry

Francesco Vista, Giovanni Iacovelli and Luigi Alfredo Grieco contributed equally to this work.

✉ Francesco Vista
francesco.vista@poliba.it

Giovanni Iacovelli
giovanni.iacovelli@poliba.it

Luigi Alfredo Grieco
alfredo.grieco@poliba.it

1 Department of Electrical and Information Engineering, Politecnico di Bari, Bari, Italy

2 Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT), Parma, Italy

Published online: 18 January 2023

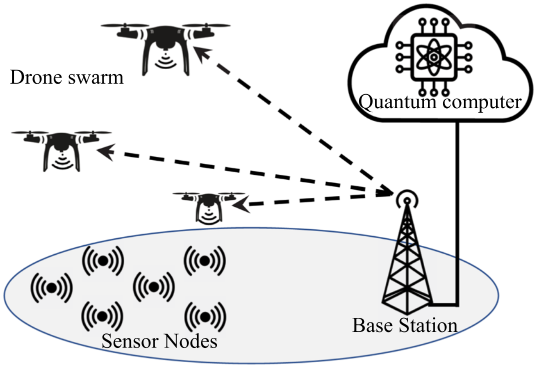


Fig. 1 Reference scenario

and academia are putting much effort in increasing the number of quantum bits (qubits) in quantum devices. Differently from classical bits, a qubit can assume a superposition of two states until a measurement takes place, thus allowing computation capabilities to exponentially grow. However, due to the decoherence principle, the number of qubits is limited. In fact, maintaining the state of a qubit is very challenging, since it requires specialized infrastructure equipped with a cooling system able to maintain a near absolute zero temperature.

Despite the challenges introduced by this powerful technology, it is employed in several application domains [2] such as (i) chemistry, (ii) machine learning, (iii) finance, and (iv) telecommunications [3]. The latter gathered the attention of the scientific community which, in the future, envisions the deployment of quantum computing devices at the edge of 6G networks to enhance service provisioning [4]. According to its inherent characteristics, quantum computing can be employed to solve binary integer programming problems.

The main algorithm used to solve this class of optimization problems is quantum annealing (QA). Similarly to simulated annealing (SA), it is inspired by the annealing process in metallurgy. In this case, instead of the temperature decrease, the strength of the transverse field is used to control the probability of quantum tunneling between states. In the end, the system remains in the lowest energy level which corresponds to the solution of the original problem. The first physical implementation was realized in 2011 by D-Wave Systems [5].

Recently, the pros and cons brought by quantum optimization have been investigated in the telecommunications field, mainly focusing on scheduling in wireless networks.

In particular, [6] investigates a scenario in which a set of sensors are organized in a tree network topology as a part of a wireless network. Several parent sink nodes are in charge of collecting and aggregating sensing data generated by their child nodes. With the aim of minimizing the overall collecting time, an optimization problem is formulated to obtain the optimal scheduling plan, while considering constraints related to interference among nodes and time division multiple access (TDMA) adopted scheme. Quantum annealing together with other methods are used to find the solution and obtained results are compared in terms of quality and computational time. With the same aim, [7] investigates a similar scenario in which a K-hop interference model is adopted. A weighted maximum independence set (WMIS) problem is formulated based on a conflict graph corresponding to possible collisions due to the activation of network nodes. The scheduling solution obtained through QA is then compared with the SA's one, where the effect of extra penalty weight adjustment is discussed. Lately, the same authors studied the same scenario in [8] but with particular focus on the adoption of Dirichlet protocol in wireless networking, showing significant improvements in the D-Wave 2X solution compared with that of its predecessor, the D-Wave II.

Although interesting, these works do not consider the channel model and the optimization of network resources. To fill the gap, [9] gives a preliminary evaluation of QA algorithm applied to resource scheduling, emphasizing its benefits and drawbacks. This work further investigates these aspects by envisioning a scenario, depicted in Fig. 1, in which a swarm of drones gathers data generated by a set of sensors while considering a widely adopted communication model for unmanned aerial vehicle (UAV). Since the system adopts the TDMA protocol, this work aims to fairly distribute network resources to sensor nodes (SNs) by employing a hybrid-quantum classical approach. As a result, a binary optimization problem is derived and then reformulated as a binary quadratic (BQ) form in order to solve it on D-Wave's quantum annealer. To evaluate its potential, a comparison with classical SA and tabu search (TS) algorithm is carried out. The resulting scheduling plans are then evaluated in terms of cost function value, computational time, and data rate. Results show that hybrid quantum-classical approach is still in its early stage. In fact, classical algorithms perform better in terms of execution time, especially in case of a low number of SNs, while no significant advantages in terms of solution quality emerge.

The rest of the work is organized as follows: Section 2 gives a brief introduction to quantum optimization. Sections 3 and 4 describe the system model and the problem formulation, respectively. Section 5 presents the obtained numerical results. Finally, Sect. 6 closes the work and draws future research perspectives.

2 Background

Quantum optimization is employed to overcome the classical limits in solving combinatorial optimization problems. In particular, QA is a metaheuristic algorithm able to find the global minima of a given problem by manipulating a set of qubits states. Such a problem needs to be formulated through a time-dependent Hamiltonian $H(t)$ defined as

$$H(t) = s(t)H_0 + (1 - s(t))H_1, \tag{1}$$

where H_0 is an initial Hamiltonian whose ground state, i.e., minimal energy configuration, is easy to find and prepare. H_1 represents the given problem and, as a consequence, its lowest energy level corresponds to the optimal solution. The adiabatic theorem of quantum mechanics states that if $s(t)$, which mathematically represents the transition function, is decreased slowly enough from 1 to 0, the system converges to a state close to the ground one of H_1 [10]. Concretely, at the beginning of the process $H(0) = H_0$, while at the end of the computation, after τ seconds, $H(\tau) = H_1$. The above process inspired the construction of D-wave's quantum processing units (QPUs), which are a physical representation of an undirected graph with a limited number of qubits and connections among them. On this basis, a QPU implementing a quantum annealer algorithm is able to find the optimal solution of a problem in BQ form, i.e., a quadratic polynomial of binary variables. Such a model can be expressed in two equivalent formulations, i.e., Ising model and quadratic unconstrained binary optimization (QUBO).

The former is commonly used in statistical mechanics and considers the solution variables as spins $s_i \in \{+1, -1\}$, which can assume two states, i.e., spin up ($\uparrow$) and spin down ($\downarrow$). Relationships between the spins, represented by couplings, are correlations or anti-correlations. The objective function defined through the Ising model is

$$\sum_{i=1}^{N} h_i s_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} J_{i,j} s_i s_j, \tag{2}$$

where N denotes the number of qubits, h_i describes the linear coefficients, i.e., qubit biases, and $J_{i,j}$ are the coupling strengths of the quadratic spin terms.

The latter is traditionally employed in computer science, since it uses binary variables, i.e., x_i, which remind classical bits. The QUBO objective function can be expressed as follows:

$$\sum_{i}^{N} Q_{i,i} x_i + \sum_{i<j}^{N} Q_{i,j} x_i x_j, \tag{3}$$

where $Q \in \mathbb{R}^{N \times N}$ is an upper triangular matrix, whose diagonal elements correspond to linear coefficients, while off-diagonal ones are the coupled coefficients. An equivalent concise matricial form is

$$x^T Q x. \tag{4}$$

The above formulation, which is used in this work, does not inherently account for the presence of constraints that have to be included by adopting specific strategies, such as penalty methods. It is worth noting that (2) and (3) are exchangeable by means of a linear transformation, i.e., $x_i = (s_i + 1)/2$.

Once the BQ problem is derived, it is necessary to map it onto the constrained topology of the QPU. This process, called embedding, is practically done by associating the linear coefficients to qubit biases and quadratic ones to coupling strengths. Furthermore, whereas the problem structure cannot be directly embedded into the QPU topology, e.g., due to the limited connection between qubits, a logical variable is represented by a chain of physical qubits. Note that a solution is consistent if all qubits in a chain have the same value. The embedding can be performed manually or by heuristic algorithms, such as MinorMiner [11].

3 System model

The mission duration T is discretized into $k = 1, \dots, K$ intervals, each one lasting δ_t seconds. A swarm of $m = 1, \dots, M$ drones, located at $\mathbf{q}_m \in \mathbb{R}^3$, hover over a set of $n = 1, \dots, N$ SNs, placed in $\mathbf{u}_n \in \mathbb{R}^3$. Moreover, it is assumed that each drone and each node are equipped with one wireless communication unit. To avoid interference among UAVs and SNs, the communications toward different drones take place on different sub-bands, by adopting the orthogonal frequency multiple access (OFDMA) scheme, and different timeslots, by employing the TDMA scheme. Therefore, the scheduling plan is described by means of a 2D binary matrix $\mathbf{x} \in \{0, 1\}^{M \times N}$, containing column vectors denoted as $\mathbf{x}_n[k] \in \{0, 1\}^{M \times 1}$ and its components defined as $x_{m,n}[k]$. Specifically, only when $x_{m,n}[k] = 1$, the mth UAV serves the nth SN. It is further assumed that all nodes are equipped with a wake-up receiver which allows to (i) recover from a sleep state to save energy and (ii) identify the associated UAV and its corresponding sub-band. Besides, the transmission power of each SN is constant and hence it is defined as $P_n \forall n$. The channel gain [12] between a UAV m and a node n, for each k, is equal to $h_{m,n} = \sqrt{\beta_0 d_{m,n}^{-\alpha}}$, where β_0 is the reference channel power gain, α is the pathloss coefficient, and $d_{m,n} = \left\| \mathbf{q}_m - \mathbf{u}_n \right\|$ is the UAV-SN distance. Therefore, the channel capacity of a UAV-SN link can be expressed as $r_{m,n} = B \log_2 \left(1 + \frac{P_n |h_{m,n}|^2}{\sigma^2} \right)$, where σ^2 denotes the noise power and B is the bandwidth. Given the mth UAV, $\mathbf{r}_n = [r_{1,n}, \dots, r_{m,n}, \dots, r_{M,n}]^\mathsf{T}$ is the column vectors containing the achievable data rates of all SNs.

4 Problem definition

To reduce the computational complexity, the proposed formulation accounts for a fixed timeslot j, which corresponds to lower the dimensionality from $K \times M \times N$ to $M \times N$.

4.1 Classical formulation

To derive the whole scheduling plan, it is necessary to solve the following problem for each timeslot:

$$(P1): \min_{\mathbf{x}[j]} \sum_{\substack{\{n,n'\}=0, \\ n \neq n'}}^{N} \left(\sum_{k=1}^{j} \mathbf{x}_n[k]^\mathsf{T} \mathbf{r}_n - \sum_{k=1}^{j} \mathbf{x}_{n'}[k]^\mathsf{T} \mathbf{r}_{n'} \right)^2 \ \textbf{s.t.}$$

$$\sum_{n=1}^{N} x_{m,n}[j] = 1, \quad \forall m : 1 \dots M, \tag{5}$$

$$\sum_{m=1}^{M} x_{m,n}[j] = 1, \quad \forall n : 1 \dots N, \tag{6}$$

$$x_{m,n}[j] \in \{0, 1\}, \quad \forall m : 1...M, n : 1...N. \tag{7}$$

Problem $(P1)$ aims to optimally allocate timeslot j to SNs, thus fairly distributing resources throughout the mission. This can be mathematically modeled as minimizing the difference between data rates for each sensor couple. It is worth noting that the objective function takes into consideration the information exchanged in previous instants. In fact, given $\{n, n'\}$

$$\Delta_{j,n,n'} \triangleq \sum_{k=1}^{j-1} \mathbf{x}_n[k]^\mathsf{T}\mathbf{r}_n - \sum_{k=1}^{j-1} \mathbf{x}_{n'}[k]^\mathsf{T}\mathbf{r}_{n'}, \tag{8}$$

is a known quantity, derived from past iterations (when $j = 1$ also $\Delta_{j,n,n'} = 0$). Therefore, an equivalent formulation of $(P1)$ is

$$(P2) : \min_{\mathbf{x}[j]} \sum_{\substack{\{n,n'\}=0,\\ n\neq n'}}^{N} \left(\mathbf{x}_n[j]^\mathsf{T}\mathbf{r}_n - \mathbf{x}_{n'}[j]^\mathsf{T}\mathbf{r}_{n'} + \Delta_{j,n,n'}\right)^2 \textbf{ s.t.} \qquad (5)(6)(7)$$

Constraint (5) imposes that in timeslot j, no more than one sensor can communicate with the same UAV. Similarly, (6) states that a drone has to serve a single SN, in instant j. The constraint (7) guarantees that the scheduling plan is composed of binary values.

Algorithm 1 Proposed scheduling optimization algorithm

1: Initialize the sensors and drones position as $\mathbf{q}_m$ and $\mathbf{u}_n$, respectively;
2: Compute channel capacity $r_{m,n}$ for each drone-sensor couple;
3: Let $\Delta_{1,n,n'} = 0$;
4: **for** k : 1 `to` K **do**
5: Solve (P3) to obtain the optimal solution $\{\mathbf{x}[k]^*\}$;
6: Compute $\Delta_{k+1,n,n'}$ as described in (8);
7: **end for**

4.2 QUBO formulation

To solve the problem by employing a QPU, the original problem $(P2)$ is mapped to QUBO form [13]. Therefore, the final objective function is defined as the Hamiltonian $H = H_A + H_B + H_C$ where

$$H_A = \sum_{\substack{\{n,n'\}=0,\\ n\neq n'}}^{N} \left(\mathbf{x}_n[j]^\mathsf{T}\mathbf{r}_n - \mathbf{x}_{n'}[j]^\mathsf{T}\mathbf{r}_{n'} + \Delta_{j,n,n'}\right)^2,$$

$$H_B = \lambda \left(1 - \sum_{n=1}^{N} x_{m,n}[j]\right)^2,$$

$$H_C = \eta \left(1 - \sum_{m=1}^{M} x_{m,n}[j]\right)^2$$

and $\{\lambda, \eta\} > 0$ as penalty factors. Note that the objective function of $(P2)$ is already in quadratic form and, hence, does not require any manipulation. On the contrary, constraints (5) and (6) have been reformulated involving the quadratic penalty method [13]. Besides, (7) is inherently addressed since quantum optimization is employed. The final unconstrained formulation of the QUBO problem is

$$(P3): \min_{\mathbf{x}[j]} \sum_{\substack{\{n,n'\}=0,\\ n\neq n'}}^{N} \left(\mathbf{x}_n[j]^{\mathsf{T}}\mathbf{r}_n - \mathbf{x}_{n'}[j]^{\mathsf{T}}\mathbf{r}_{n'} + \Delta_{j,n,n'}\right)^2$$
$$+ \lambda \left(1 - \sum_{n=1}^{N} x_{m,n}[j]\right)^2 + \eta \left(1 - \sum_{m=1}^{M} x_{m,n}[j]\right)^2,$$

that can be implemented on a quantum system to be solved, as described in Algorithm 1.

5 Numerical results and discussion

In this section, a simulation campaign has been conducted to evaluate the effectiveness of the proposed formulation by employing the D-wave leap hybrid solver, which uses a hybrid quantum-classical approach. This solver is suitable for problems with a large number of variables that cannot be mapped directly into QPU's topology. In particular, a classical process divides the original problem into sub-problems that are dispatched to the QPU and to the cloud's classical computing capabilities. The obtained results are compared with two classical optimization algorithms, i.e., SA and TS, implemented in the D-Wave Python library, running on a computer with an Intel i5 6200U @ 2.8 GHz and 4 GB of RAM.

For the D-wave leap hybrid solver, default parameters have been adopted, e.g., number of reads set to 100. Besides, the minimum penalty factors have been chosen such that no improvement of the solution is obtained, i.e., $\lambda = 10^{17}$, $\eta = 10^2$. The mission time T has been split into $K = 60$ timeslot of $\delta_t = 1$ second each. Furthermore, $M = 4$ drones are deployed in $[15\ 15\ 80]^T$, $[20\ 70\ 100]^T$, $[75\ 20\ 110]^T$, and $[80\ 80\ 90]^T$, serving $N = \{25, 50, 75, 100\}$ SNs uniformly distributed over a 100x100 m area. As for the transmission, $B = 1$ MHz, $P_n = 10$ mW $\forall n$, $\alpha = 2$, $\sigma^2 = N_0 B$, and $N_0 = -174$ dBm/Hz [14].

Given the scheduling plan $\mathbf{x}$, obtained as the solution of problem $(P3)$, it is possible to compute the sum-rates of each sensor at the end of the mission.

Thanks to the proposed formulation, as can be seen from mean and standard deviation reported in Table 1, UAVs are able to fairly serve nodes regardless the number of SNs and algorithm used, without showing any sensible difference. As a matter of fact,

Table 1 Sum-rate means and standard deviations of the algorithms

		Algorithms					
		H		SA		TS	
		Mean	Std	Mean	Std	Mean	Std
SNs	25	263.09	12.074	264.41	12.783	262.55	11.697
	50	132.38	10.556	132.53	10.748	132.31	10.417
	75	87.72	13.084	87.86	10.769	87.23	10.41
	100	65.69	13.084	65.88	13.129	65.45	12.961

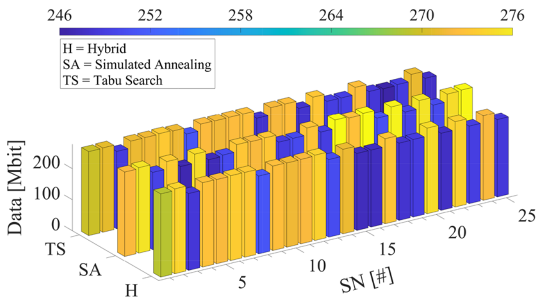


Fig. 2 Acquired data at the end of the mission, with $N = 25$

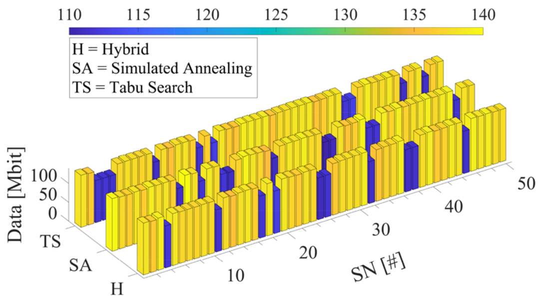


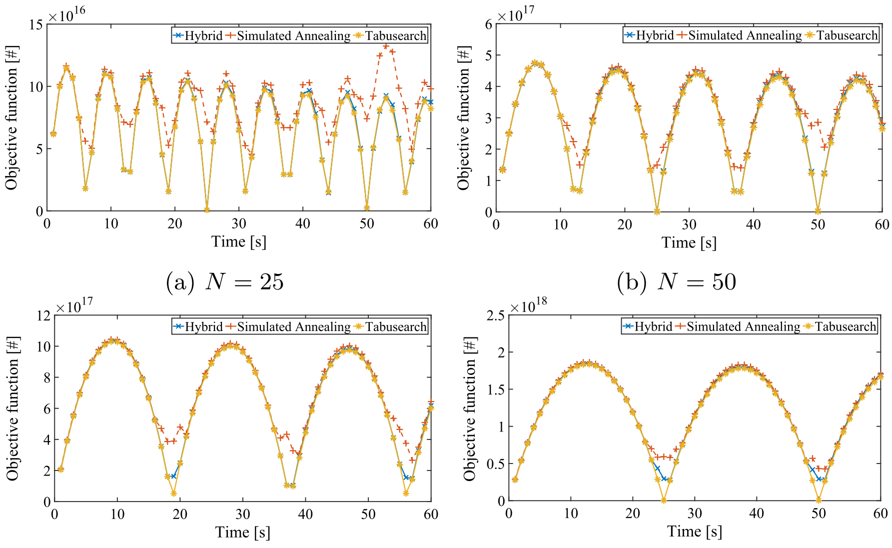
Fig. 3 Acquired data at the end of the mission, with $N = 50$

when the number of SNs approaches the number of drones, the amount of gathered data increases since a drone serves less sensors during the whole mission, vice versa when $M \ll N$ the sum-rate decreases. Indeed, in the first configuration, the swarm is able to collect ~ 260 Mbits, while in the last one just ~ 65 Mbits.

To provide further insights, for each sensor, the sum-rates in case of $N = 25$ and $N = 50$ are shown in Figs. 2 and 3, respectively. Although different amounts for each SN are exhibited, the transmission fairness is clearly achieved, regardless the employed algorithm.

A comparison among the different objective function curves for different algorithms is presented in Fig. 4. When 25 SNs are considered, the classical SA algorithm performs worse than the hybrid approach and TS algorithm, which instead provide comparable results. As the number of SNs increases, i.e., $M \ll N$, a difference is still present although irrelevant.

Finally, a thorough comparison of the algorithms' execution time, to solve the formulated optimization problem, is hereby analyzed. For what concerns the hybrid



(a) $N = 25$ (b) $N = 50$

(c) $N = 75$ (d) $N = 100$

Fig. 4 Comparison of objective function curves with different number of SNs

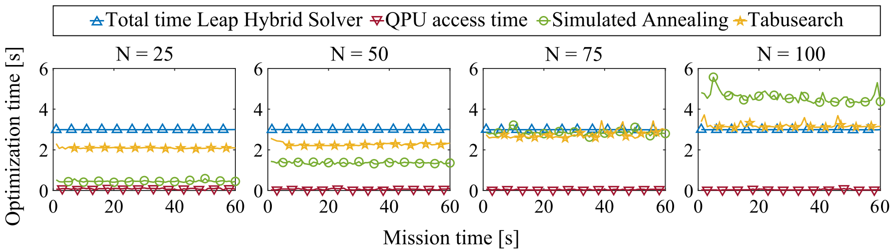


Fig. 5 Comparison of execution time for each solver

solver, the total time and the QPU access time are reported separately [15, 16]. The latter is a portion of overall hybrid solver time and consists of (i) a one-time setup procedure to prepare the QPU and (ii) the sampling time. It should be noted that the embedding procedure period, the network latency and the queuing time, which all take approximately 4 s, are not included in this analysis [17].

As depicted in Fig. 5, regardless the number of SN, the hybrid solver takes about 3 s to complete the process. Instead, for $N = 25$ and $N = 50$, classical algorithms perform slightly better in terms of execution time, which confirms the results obtained in [15–17]. As the number of sensors increases, the execution time of SA and TS increases as well. In particular, for $N = 75$, classical algorithms and hybrid solver give comparable results. This trend remains for $N = 100$, except for SA which performs worse, i.e., ~ 5 s.

6 Conclusions

In this work, scheduling optimization is employed to fairly allocate channel resources of a UAV-enabled Internet of Things (IoT) network. A combinatorial problem stem from the proposed formulation, which is first encoded into QUBO form and then solved by (i) hybrid quantum-classical approach, (ii) SA, and (iii) TS. According to the results, quantum optimization is still in its infancy. In fact, no significant gain is observed in solution quality. From a execution time perspective, instead, classical algorithms take less or comparable time with respect to the hybrid solver, with the only exception of SA in case of large number of sensors. More research is needed to overcome the limitations brought by quantum computing applied to optimization problems. Future works will investigate the joint optimization of transmission scheduling plan, multiple drones' trajectories, as well as the correspondent energy consumption.

Acknowledgements This work has been supported by the PRIN project no. 2017NS9FEY entitled "Real-time Control of 5G Wireless Networks: Taming the Complexity of Future Transmission and Computation Challenges" funded by the Italian MIUR, the project entitled "The house of emerging technologies of Matera (CTEMT)" funded by the Italian MISE, the Italian MIUR PON projects AGREED (ARS01_00254), and the Warsaw University of Technology within IDUB programme (Contract No. 1820/29/Z01/POB2/2021).

Data availability Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

1. Gyongyosi, L., Imre, S.: A survey on quantum computing technology. Comput. Sci. Rev. **31**, 51–71 (2019)
2. Caleffi, M., Cacciapuoti, A.S., Bianchi, G.: Quantum Internet: from communication to distributed computing! In: Proceedings of the 5th ACM International Conference on Nanoscale Computing and Communication (2018)
3. Kim, M., Venturelli, D., Jamieson, K.: Leveraging quantum annealing for large MIMO processing in centralized radio access networks. In: Proceedings of the ACM Special Interest Group on Data Communication, pp. 241–255 (2019)
4. Vista, F., Musa, V., Piro, G., Grieco, L.A., Boggia, G.: Network intelligence with quantum computing in 6g and b6g: Design principles and future directions. In: 2021 IEEE Globecom Workshops (GC workshops), pp. 1–6 (2021)
5. Johnson, M.W., Amin, M.H., Gildert, S., Lanting, T., Hamze, F., Dickson, N., Harris, R., Berkley, A.J., Johansson, J., Bunyk, P.: Quantum annealing with manufactured spins. Nature **473**(7346), 194–198 (2011)
6. Ishizaki, F.: Computational method using quantum annealing for TDMA scheduling problem in wireless sensor networks. In: 2019 13th International Conference on Signal Processing and Communication Systems (ICSPCS), pp. 1–9 (2019)
7. Wang, C., Chen, H., Jonckheere, E.: Quantum versus simulated annealing in wireless interference network optimization. Sci. Rep. **6**(1), 1–9 (2016)
8. Wang, C., Jonckheere, E.: Simulated versus reduced noise quantum annealing in maximum independent set solution to wireless network scheduling. Quantum Inf. Process. **18**(1), 1–25 (2019)
9. Vista, F., Iacovelli, G., Grieco, L.A.: Quantum scheduling optimization for UAV-enabled IoT networks. In: Proceedings of the CoNEXT Student Workshop. CoNEXT-SW '21, pp. 19–20 (2021)

10. Hauke, P., Katzgraber, H.G., Lechner, W., Nishimori, H., Oliver, W.D.: Perspectives of quantum annealing: methods and implementations. Rep. Prog. Phys. **83**(5), 054401 (2020)
11. Cai, J., Macready, W.G., Roy, A.: A practical heuristic for finding graph minors. arXiv:1406.2741 (2014)
12. Zeng, Y., Zhang, R.: Energy-efficient UAV communication with trajectory optimization. IEEE Trans. Wireless Commun. **16**(6), 3747–3760 (2017)
13. Glover, F., Kochenberger, G., Du, Y.: Quantum Bridge Analytics I: a tutorial on formulating and using QUBO models. 4OR 17(4), 335–371 (2019)
14. Iacovelli, G., Grieco, L.A.: Drone swarm as mobile relaying system: a hybrid optimization approach. IEEE Trans. Veh. Technol. **70**(11), 12272–12277 (2021)
15. Feld, S., Roch, C., Gabor, T., Seidel, C., Neukart, F., Galter, I., Mauerer, W., Linnhoff-Popien, C.: A hybrid solution method for the capacitated vehicle routing problem using a quantum annealer. Front. ICT **6**, 13 (2019)
16. Hussain, H., Javaid, M.B., Khan, F.S., Dalal, A., Khalique, A.: Optimal control of traffic signals using quantum annealing. Quantum Inf. Process. **19**(9), 1–18 (2020)
17. Kizilirmak, R.C.: Quantum annealing approach to NOMA signal detection. In: 2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), pp. 1–5 (2020)