Quantum Heuristic Algorithms for Hard Planning Problems from Aerospace Research

Thorge Müller, Elisabeth Lobe, Tobias Stollenwerk

German Aerospace Center (DLR), Cologne

September 16, 2019
Helmholtz-Triumf-Workshop
Hamburg

DLR

Content

- Quantum Computing at the German Aerospace Center
- Quantum Heuristic Algorithms for Flight Gate Assignment

Quantum Computing at DLR - Strategy

- Our Mission:

  How improve aerospace research with QC?

- Our Approach:

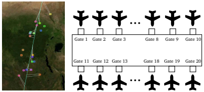
Quantum Computing at DLR - Strategy

- Our Mission:

  How improve aerospace research with QC?

- Our Approach:

Aerospace Research



Quantum Computing at DLR - Strategy

- Our Mission:

  How improve aerospace research with QC?

- Our Approach:

Fundamental Research

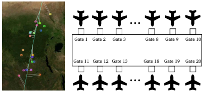
SAARLAND UNIVERSITY

Universität zu Köln

JÜLICH FORSCHUNGSZENTRUM

DLR

Aerospace Research



Quantum Computing at DLR - Strategy

- Our Mission:

  How improve aerospace research with QC?

- Our Approach:

Fundamental Research

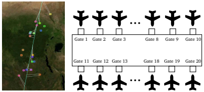
SAARLAND UNIVERSITY

Universität zu Köln

JÜLICH FORSCHUNGSZENTRUM

DLR

Aerospace Research

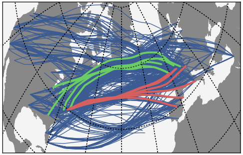


NASA

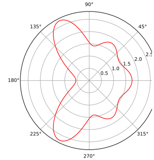
Quantum Artificial Intelligence Lab

DLR

Quantum Computing at DLR - Topics



Applications for Q-Annealing (DLR/NASA)



Applications for gate-based QC

```
# test polynomial class
class testPolynomial(unittest.TestCase):
    def setUp(self):
        # qubo1 : 5 x1 + 6 x2 + 3 x1 x3 + x4 + 3 x2 x4 +
        self.Q1 = polynomial.Polynomial({(1,): 5, (2,):
        # qubo2 : 7 x5 + x2 x4 + 3 x3 x4 + x2
        self.Q2 = polynomial.Polynomial({(2,): 1, (2, 4)
        # qubo3 : 7 x1 + 2 x2 + 3 x3 + x4
        self.Q3 = polynomial.Polynomial({(1,): 7, (2,):

        self.Q4 = polynomial.Polynomial({(('x', 1),): 7,

        self.filename_h5 = testdir + "/test_qubo.h5"
        self.filename_txt = testdir + "/test_qubo.txt"

    def testEqual(self):
        Q = polynomial.Polynomial({(1,): 7, (2,): 2, (3,
        self.assertFalse(Q == self.Q1)
```

Software for QC

Quantum Compiling

Quantum Annealing process

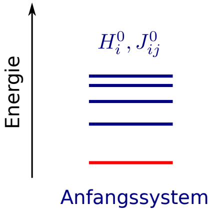
- How we reach the target state?

Quantum Annealing process

- How we reach the target state?
- solution: adiabatic development

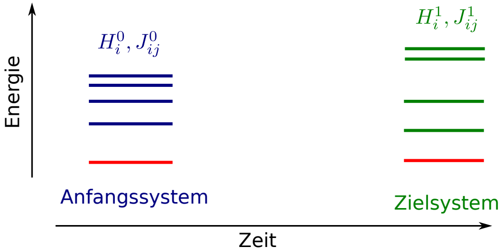
Quantum Annealing process

- How we reach the target state?
- solution: adiabatic development
  1. prepare the system in an known inital ground state



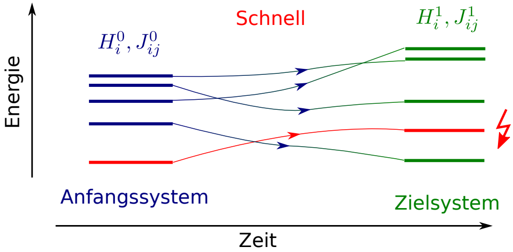
Quantum Annealing process

- How we reach the target state?
- solution: adiabatic development
    1. prepare the system in an known inital ground state



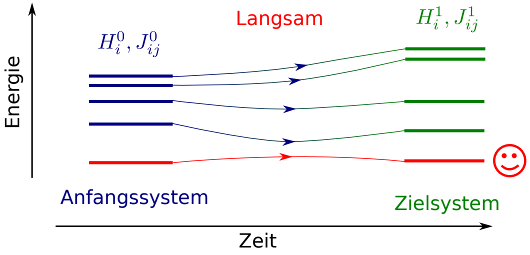
Quantum Annealing process

- How we reach the target state?
- solution: adiabatic development
    1. prepare the system in an known inital ground state



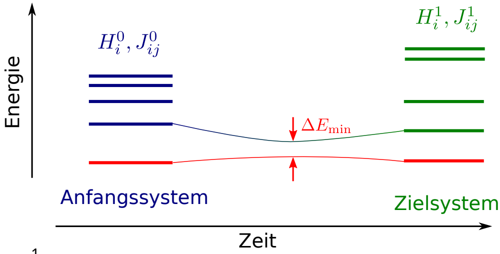
Quantum Annealing process

- How we reach the target state?
- solution: adiabatic development
  1. prepare the system in an known inital ground state
  2. change *slowly* towards the target sytsem



Quantum Annealing process

- How we reach the target state?
- solution: adiabatic development
  1. prepare the system in an known inital ground state
  2. change *slowly* towards the target sytsem

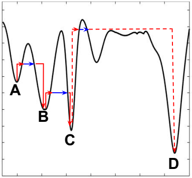


- runtime $\propto \frac{1}{\Delta E_{\min}}$

runtime of Quanum Annealers

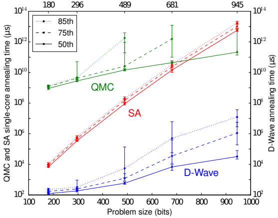
there are hints of the supremacy of quantum annealers against classcial approaches

- problems with **high** and **narrow** barriers
- quantum tunneling through barriers



open questions:

- Is there a supremacy for real use cases?
- are there problems with a better scaling?



Denchev et. al., Google, arXiv:1512.02206

Flight Gate Assignment - Problem Size

A day at Frankfurt Airport

- about 1300 aircraft movements (arrival and departure)
- more than 90% are passenger flights
- more than 170000 passengers
- about 60% transfer passengers
- 278 gates

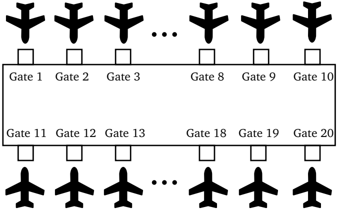
Frankfurt Airport

DLR

Flight Gate Assignment - Decision Variable

Flight i

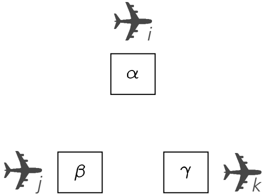
Gate α



Decision variable

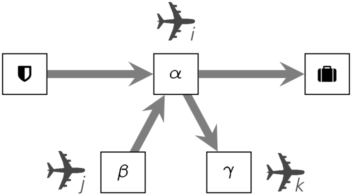
$$x_{i\alpha} = \begin{cases} 1, & \text{if flight } i \text{ is assigned to gate } \alpha, \\ 0, & \text{otherwise} \end{cases}$$

Flight Gate Assignment - Cost Function



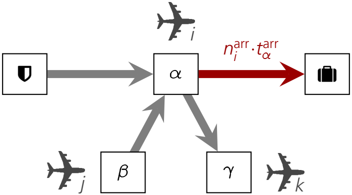
DLR

Flight Gate Assignment - Cost Function



Minimizing the total transfer time with the cost function

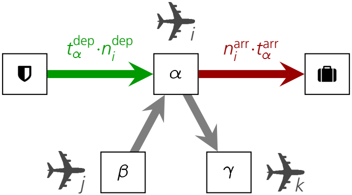
Flight Gate Assignment - Cost Function



Minimizing the total transfer time with the cost function

$$C = \sum_{i\alpha} n_i^{arr} t_\alpha^{arr} x_{i\alpha}$$

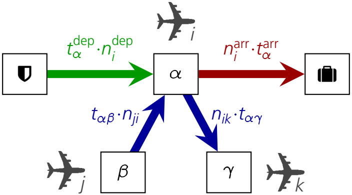
Flight Gate Assignment - Cost Function



Minimizing the total transfer time with the cost function

$$C = \sum_{i\alpha} n_i^{arr} t_\alpha^{arr} x_{i\alpha} + \sum_{i\alpha} n_i^{dep} t_\alpha^{dep} x_{i\alpha}$$

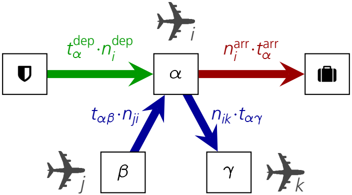
Flight Gate Assignment - Cost Function



Minimizing the total transfer time with the cost function

$$C = \sum_{i\alpha} n_i^{arr} t_\alpha^{arr} x_{i\alpha} + \sum_{i\alpha} n_i^{dep} t_\alpha^{dep} x_{i\alpha} + \sum_{ij\alpha\beta} n_{ij} t_{\alpha\beta} x_{i\alpha} x_{j\beta}$$

Flight Gate Assignment - Cost Function



Minimizing the total transfer time with the cost function

$$C = \sum_{i\alpha} n_i^{arr} t_\alpha^{arr} x_{i\alpha} + \sum_{i\alpha} n_i^{dep} t_\alpha^{dep} x_{i\alpha} + \sum_{ij\alpha\beta} n_{ij} t_{\alpha\beta} x_{i\alpha} x_{j\beta}$$

$\rightarrow$ Quadratic Assignment problem

Flight Gate Assignment - Constraints as Penalty Terms

- One gate per flight: $\forall i: \ \sum_{\alpha} x_{i\alpha} = 1$:

Flight Gate Assignment - Constraints as Penalty Terms

- One gate per flight: $\forall i: \ \sum_\alpha x_{i\alpha} = 1$:

$$Q_C = \lambda_C \sum_i \left(\sum_\alpha x_{i\alpha} - 1 \right)^2$$

Flight Gate Assignment - Constraints as Penalty Terms

- One gate per flight: $\forall i: \ \sum_\alpha x_{i\alpha} = 1$:

$$Q_C = \lambda_C \sum_i \left(\sum_\alpha x_{i\alpha} - 1 \right)^2$$

- No arrival before departure at the same gate

$$x_{i\alpha} \cdot x_{j\alpha} = 0 \quad \forall (i,j) \in F,\ \forall \alpha$$

Flight Gate Assignment - Constraints as Penalty Terms

- One gate per flight: $\forall i: \sum_\alpha x_{i\alpha} = 1$:

$$Q_C = \lambda_C \sum_i \left(\sum_\alpha x_{i\alpha} - 1 \right)^2$$

- No arrival before departure at the same gate

$$x_{i\alpha} \cdot x_{j\alpha} = 0 \quad \forall (i,j) \in F, \forall \alpha$$

with F : set of forbidden flight pairs

Flight Gate Assignment - Constraints as Penalty Terms

- One gate per flight: $\forall i: \sum_\alpha x_{i\alpha} = 1$:

$$Q_C = \lambda_C \sum_i \left(\sum_\alpha x_{i\alpha} - 1 \right)^2$$

- No arrival before departure at the same gate

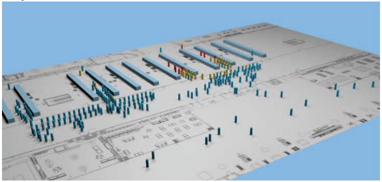
$$x_{i\alpha} \cdot x_{j\alpha} = 0 \quad \forall (i,j) \in F, \forall \alpha$$

with F : set of forbidden flight pairs

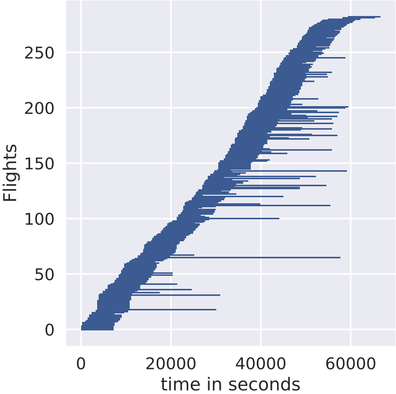
$$Q_T = \lambda_T \sum_\alpha \sum_{(i,j) \in F} x_{i\alpha} x_{j\alpha}$$

Flight Gate Assignment - Problem Instances

- Time data from agent based simulations (DLR-FW)
- Flight schedule from a mid-sized German airport
- Extract small but characteristic problem instances

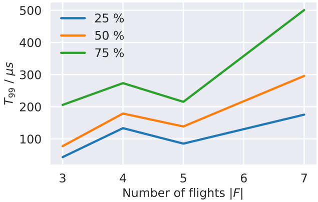


M. Jung et. al.



Flight Gate Assignment - Annealing Results

- Time-to-solution with 99% probability (p: success probability) grows with problem size
- Main Result: Success probability suppressed by increased dynamical range C_{Ising} as a result of penalty terms

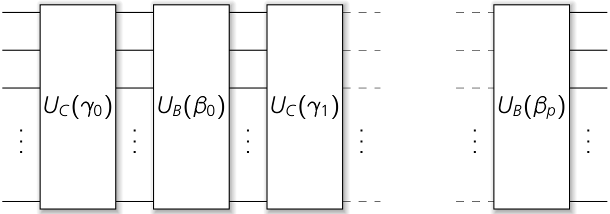


QA for Flight Gate Assignment - Summary and Outlook

- Flight gate assignment is amenable to QA
- Precision issues due to constraints
- Mitigate limited precision by bin-packing
- Open questions:
  - Are these problems hard for classical solvers?
  - How would larger problems perform?
  - Flight gate assignment with QAOA including constrained drivers
- Paper at arXiv:1811.09465

QAOA for Constraint Optimization

init state $|\psi_0\rangle$



final state $|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle$

$= e^{i\gamma_0 C}$ $= e^{i\beta_0 B}$

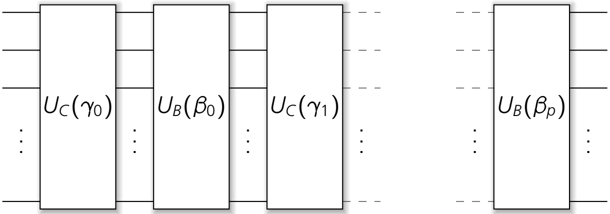
C: Cost function

B: Mixer

Hadfield et. al., arXiv:1709.03489

QAOA for Constraint Optimization

init state
$|\psi_0\rangle$



final state
$|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle$

$= e^{i\gamma_0 C}$ $\quad = e^{i\beta_0 B}$

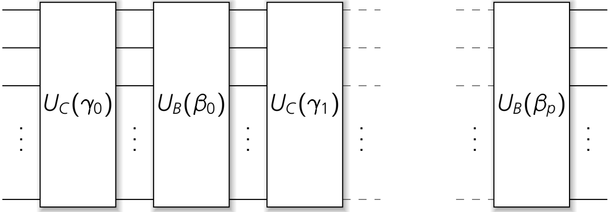
C: Cost function
B: Mixer

Get expectation value $\langle \boldsymbol{\gamma}, \boldsymbol{\beta} | C | \boldsymbol{\gamma}, \boldsymbol{\beta} \rangle$ through multiple measurements

Hadfield et. al., arXiv:1709.03489

QAOA for Constraint Optimization

init state
$|\psi_0\rangle$



final state
$|\boldsymbol{\gamma}, \boldsymbol{\beta}\rangle$

$= \; e^{i\gamma_0 C}$ $\quad = \; e^{i\beta_0 B}$

C: Cost function
B: Mixer

Get expectation value $\langle \boldsymbol{\gamma}, \boldsymbol{\beta} | C | \boldsymbol{\gamma}, \boldsymbol{\beta} \rangle$ through multiple measurements

Optimize classically

Hadfield et. al., arXiv:1709.03489

QAOA for Constraint Optimization

Find suitable mixer B

- That keeps valid states valid
- That explores the whole space

Example:

$$\sum_{i\alpha} x_{i\alpha} = 1$$

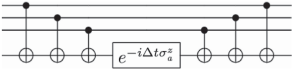
Use SWAP mixer:

$$\begin{matrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{matrix} \Rightarrow \begin{matrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{matrix}$$

Hadfield et. al., arXiv:1709.03489

QAOA Challenges for Real QC Hardware

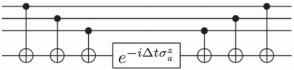
- Circuit synthesis for quantum algorithms



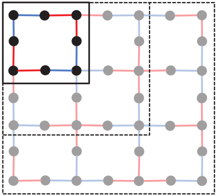
Circuit synthesis

QAOA Challenges for Real QC Hardware

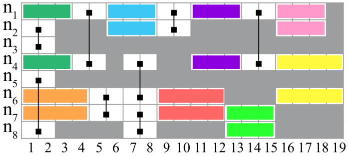
- Circuit synthesis for quantum algorithms
- Mapping quantum circuit to hardware with limited connectivity



Circuit synthesis



Limited connectivity on QC chips



QAOA compilation with temporal planners (arXiv:1705.08927)

Thank You