

Funktionsweise eines D-Wave Quantum Annealers

-2	2	2	2	2	0	0	2	0	2	0	0	2
0	-2	2	2	2	0	0	2	0	2	0	2	0
0	0	-2	2	2	2	0	2	0	0	0	2	0
0	0	0	-2	2	2	2	0	2	2	0	0	2
0	0	0	0	-2	2	2	2	0	0	2	0	2
0	0	0	0	0	-2	2	2	2	0	0	2	2
0	0	0	0	0	0	-2	2	2	2	0	0	2
0	0	0	0	0	0	0	-2	2	2	2	0	2
0	0	0	0	0	0	0	0	-2	2	2	2	0
0	0	0	0	0	0	0	0	0	-2	2	2	2
0	0	0	0	0	0	0	0	0	0	-2	2	2
0	0	0	0	0	0	0	0	0	0	0	-2	2
0	0	0	0	0	0	0	0	0	0	0	0	-2

Hamiltonian (4-Damenproblem)

```
from __future__ import print_function
import numpy as np
from dwave.system.samplers import DWaveSampler
from dwave.system.composites import EmbeddingComposite

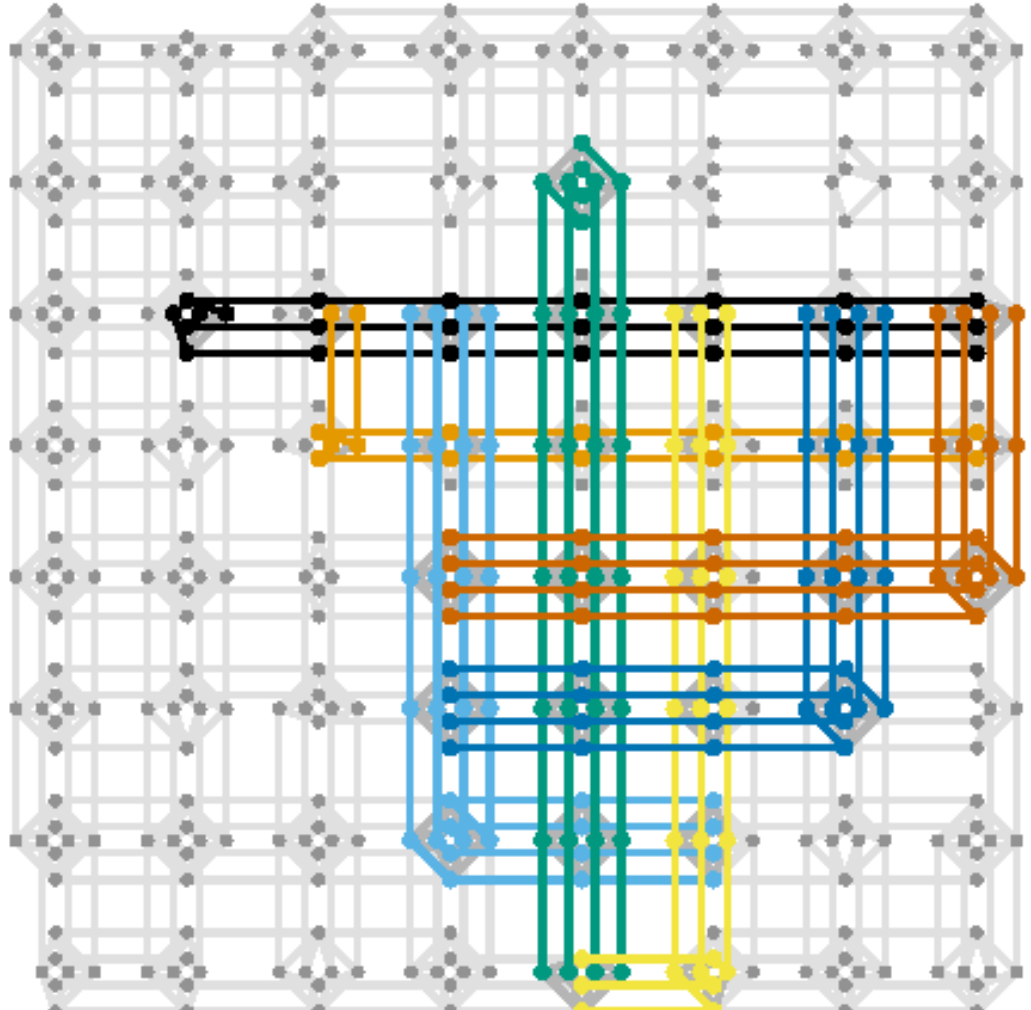
# Laden der Matrix
qubomatrix = np.loadtxt('qubomatrix.txt')
print('Loaded matrix:\n', qubomatrix, '\n')

# Konvertieren in QUBO Form
qubo = {(i,i):0.0 for i in range(len(qubomatrix))}
for index,value in np.ndenumerate(qubomatrix):
    if value != 0:
        qubo[index] = value
print('Converted matrix into QUBO for D-Wave:\n', qubo, '\n')

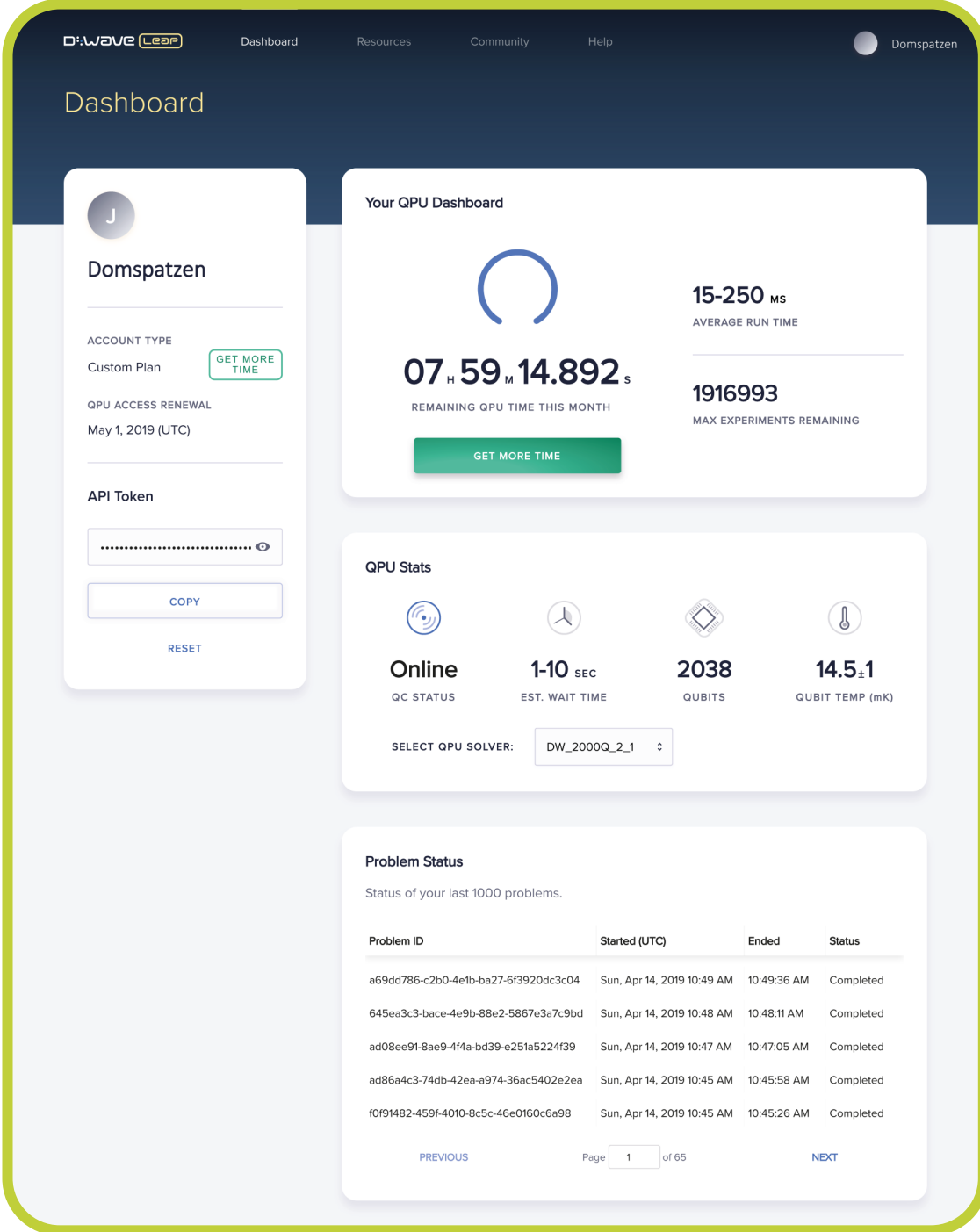
# Erstellen des Chimera-Graphen und Rechnen auf dem Quantencomputer
sampler = EmbeddingComposite(DWaveSampler())
response = sampler.sample_qubo(qubo, num_reads=10000, annealing_time=20)
print('Response from the D-Wave:\n', response, '\n')

# Speichern der Ergebnisse in results.txt
with open('results.txt','w') as file:
    file.write('energy\tnum_occurrences\tsample\n')
    for sample,energy,num_occurrences,cbf in response.record:
        file.write('%f\t%d\t%s\n' % (energy,num_occurrences,sample))
print('Saved results in results.txt')
```

Datenübermittlung an den Quantencomputer



Chimera-Graph mit 7 logischen Qubits



D-Wave Leap Portal

Ein Quantencomputer arbeitet mit sog. **Quantenbits**, die sich in einer **Superposition** der Werte 0 und 1 befinden und erst beim Auslesen einen der beiden Werte annehmen. In einem adiabatischen Quantencomputer werden Qubits mit Stromkreisen realisiert, wobei die Richtung des Stroms dem Wert des Qubits entspricht (siehe Abb. 1, Abb. 2). Qubits werden durch gewichtete Gleich- bzw. Ungleich-Koppler miteinander verbunden, so dass gekoppelte Qubits bevorzugt den gleichen oder entgegengesetzten Wert annehmen. Die **Stärke der Koppelung** bestimmt der Programmierer - sie entspricht letztlich dem „**Programmcode**“.

-8.000000	171	[0010100000010100]
-8.000000	6794	[0100000110000010]
-6.000000	2	[1000100000010100]
-6.000000	115	[0001000010000010]
-6.000000	7	[0010000000010100]
-6.000000	10	[1000000101000000]
-6.000000	6	[1010000000010100]
-6.000000	12	[1000000101000010]
-6.000000	128	[0100000100000010]
-6.000000	12	[0010100000010000]
-6.000000	26	[0001100000010100]
-6.000000	1	[0010100000000100]
-6.000000	16	[0001100000000100]
-6.000000	154	[0001010000000010]
-6.000000	12	[1000000000010100]
-6.000000	51	[1100000100000010]
-6.000000	11	[0000100000010100]
-6.000000	25	[1000001000010100]
-6.000000	87	[0001010010000010]
-6.000000	34	[1000000100000010]
-6.000000	1	[0011100000000100]

Ergebnisse für das 4-Damenproblem

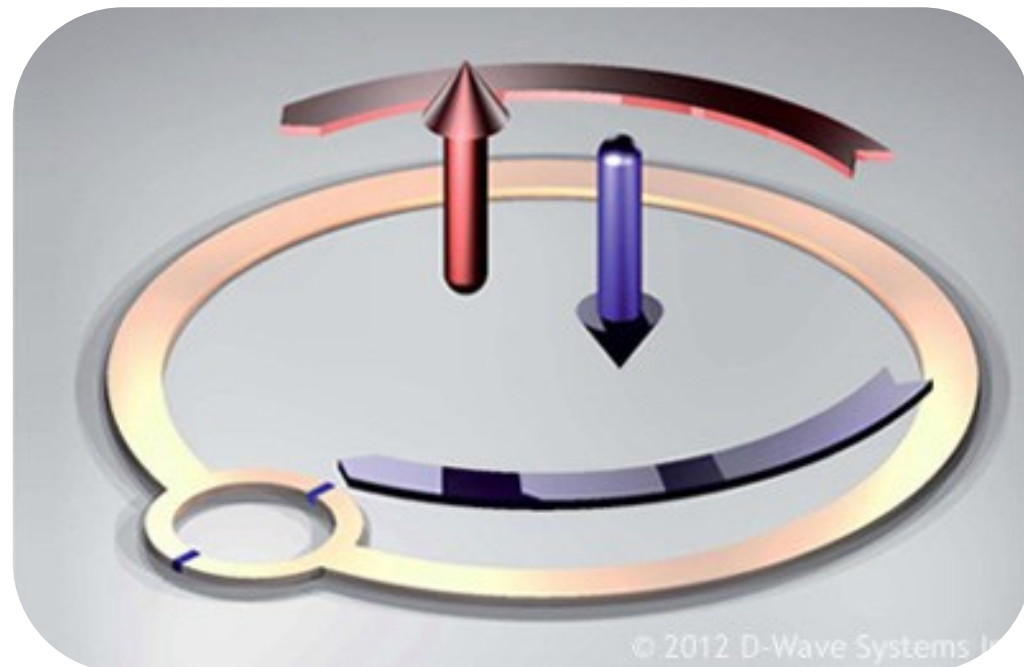


Abb. 1: Darstellung eines Qubits

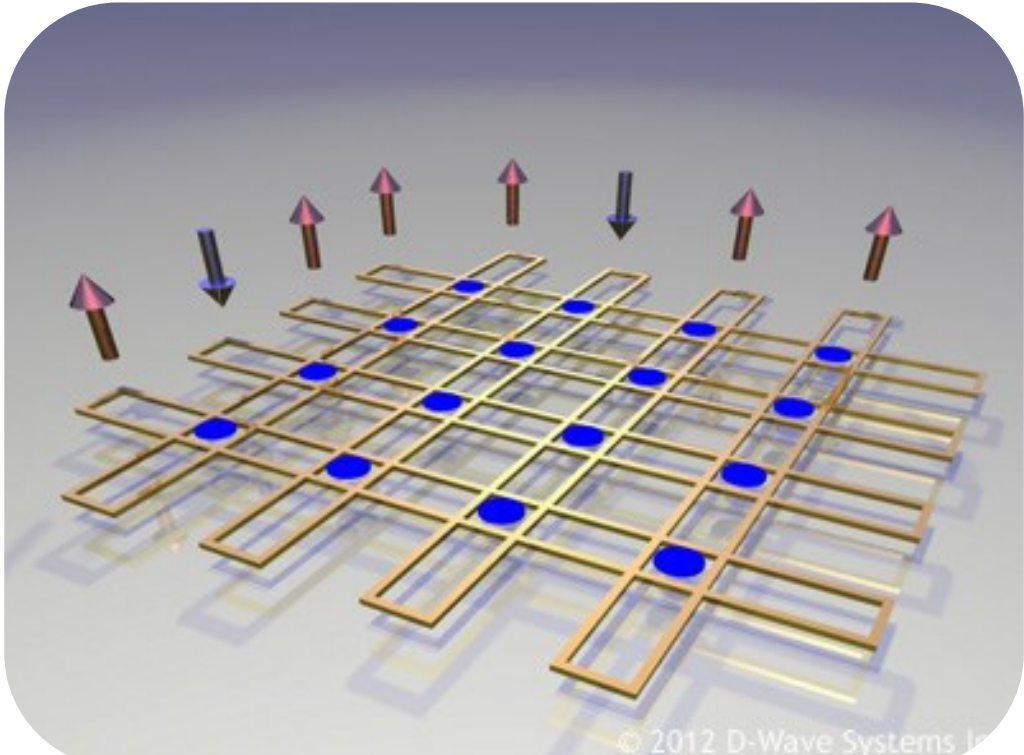


Abb. 2: Anordnung der Qubits im D-Wave Quantum-Annealer

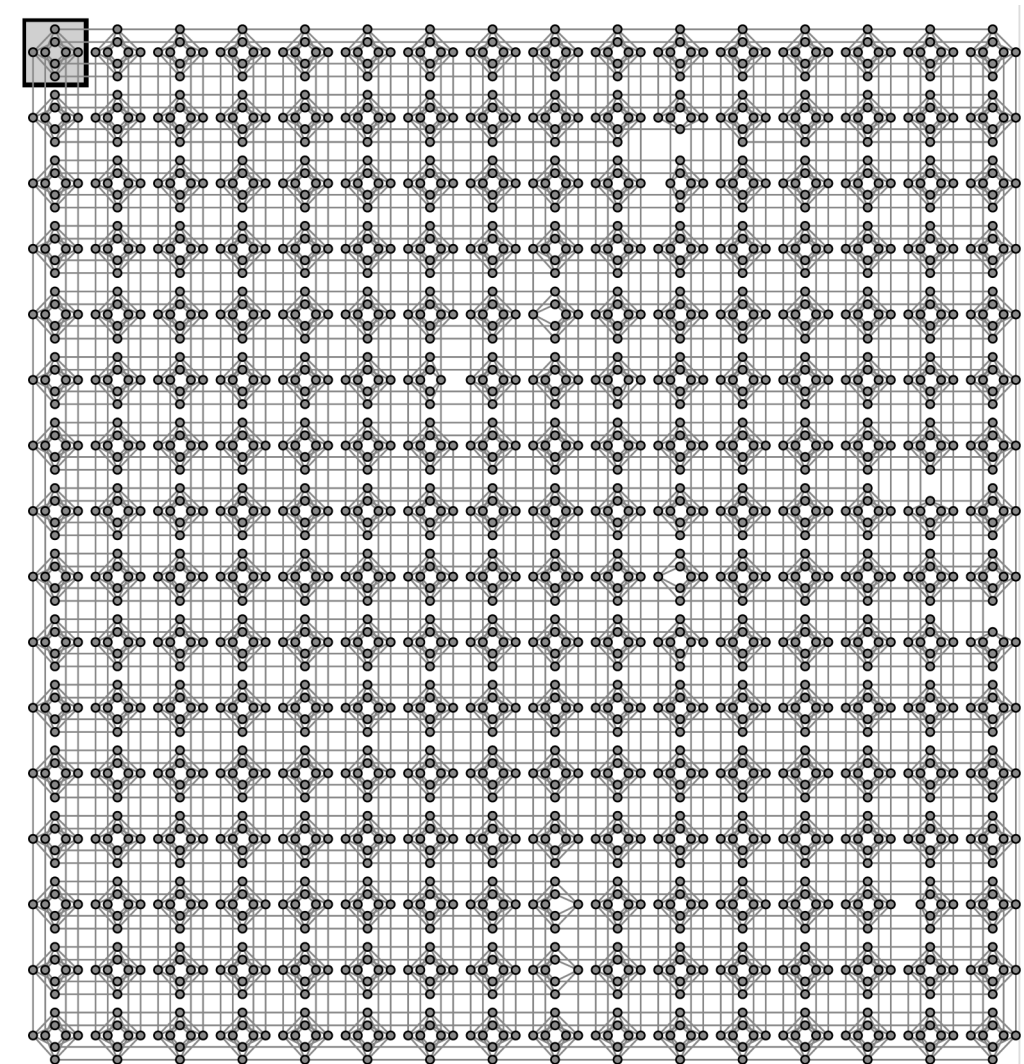
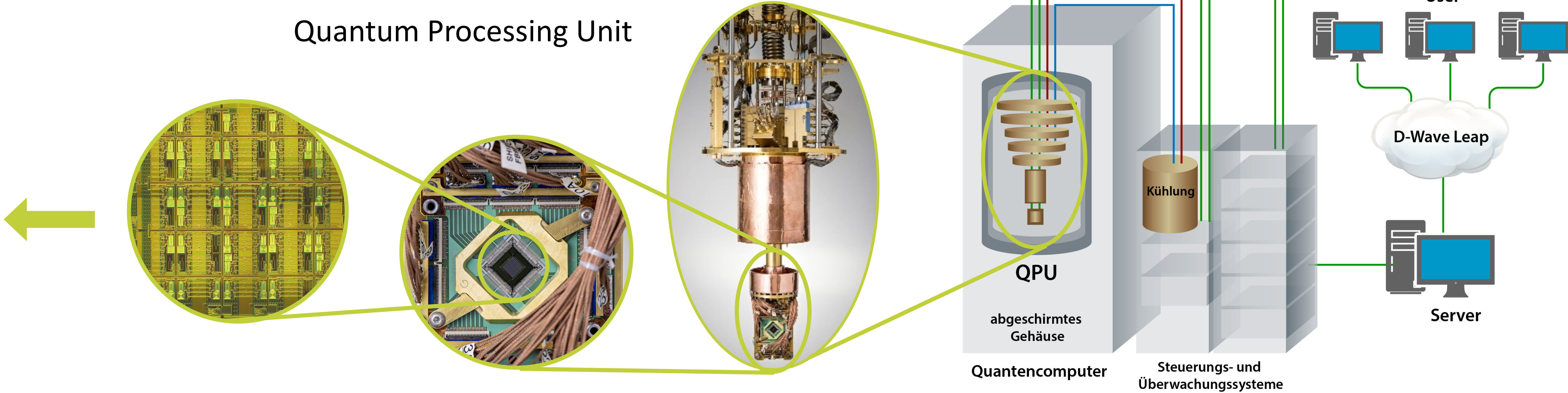


Abb. 3: Vollständiger Chimera-Graph mit 2018 physikalischen Qubits

Quantum Processing Unit



Der **Chimera-Graph** veranschaulicht, welche physikalischen Qubits auf der QPU (Quantum Processing Unit) miteinander gekoppelt wurden (siehe Abb. 3, Abb. 4). Da nicht jedes Qubit mit jedem gekoppelt werden kann, verbindet man Gruppen von **physikalischen Qubits** zu **logischen Qubits**. Diese nehmen beim Auslesen mit hoher Wahrscheinlichkeit den gleichen Wert an und können mit allen anderen logischen Qubits gekoppelt werden.

Die Stärke der Koppelungen ergibt sich aus der Hamiltonmatrix, die das zu lösende Problem mathematisch beschreibt. Nach dem Koppeln der Qubits beginnt der sog. **Anneal**, d.h. das „Ausführen des Programms“. Anfangs befinden sich alle Qubits in einer 0/1 Superposition. Dieser Zustand stellt das Energieminimum des sog. **Initial-Hamiltonians** dar. Im Laufe des Anneals wird dann der **Problem-Hamiltonian** „eingblendet“ und der Initial-Hamiltonian ausblendet. Zu Beginn ist das Grundenergieniveau, also das Energieminimum der „Mischung“ von Initial- und Problem-Hamiltonian, von den anderen Energieniveaus, die eine schlechtere Lösung des Problems darstellen, weit entfernt. Durch das Einblenden des Problem-Hamiltonians nähert sich das Grundenergieniveau den höheren Niveaus, wodurch die Wahrscheinlichkeit, dass man in ein schlechteres Energieniveau gelangt, steigt (siehe Abb. 5). Am Ende des Anneals werden die Werte der Qubits gemessen.

Theoretisch wird bei einem **adiabatischen Quantencomputer** d.h. bei unendlich langsamen Übergang zwischen beiden Hamiltonians immer das beste Ergebnis gefunden, da das Quantensystem aufgrund von **Tunneleffekten**, also der Untertunnelung von Bergen in der Energielandschaft, im niedrigsten Energiezustand bleibt (siehe Abb. 6). In der Praxis reichen Annealing-Zeiten von wenige Mikrosekunden aus, um ein gutes Ergebnis zu finden. Um die Wahrscheinlichkeit für ein gutes Ergebnis zu erhöhen, führt man ca. 10.000 Anneals nacheinander durch.

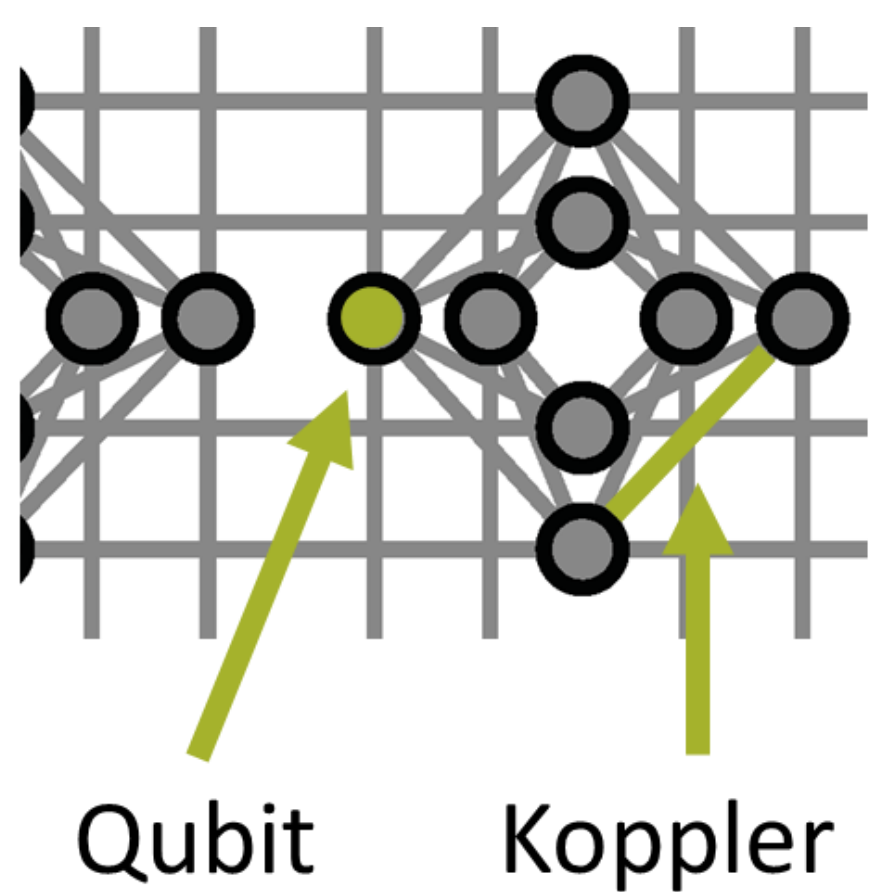


Abb. 4: Kopplung physikalischer Qubits auf dem Chimera-Graph

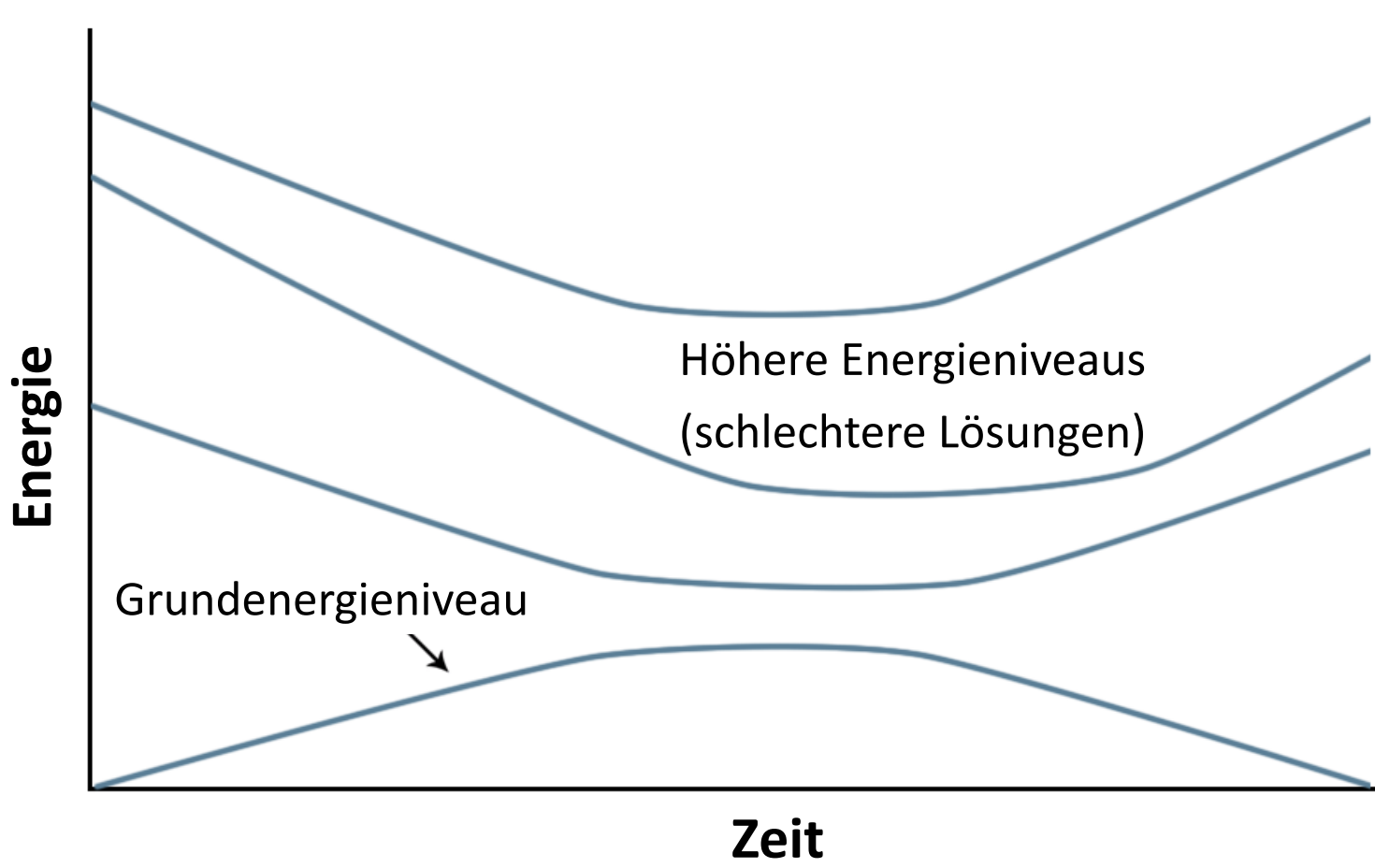


Abb. 5: Unterschiedliche Energieniveaus während eines Anneals

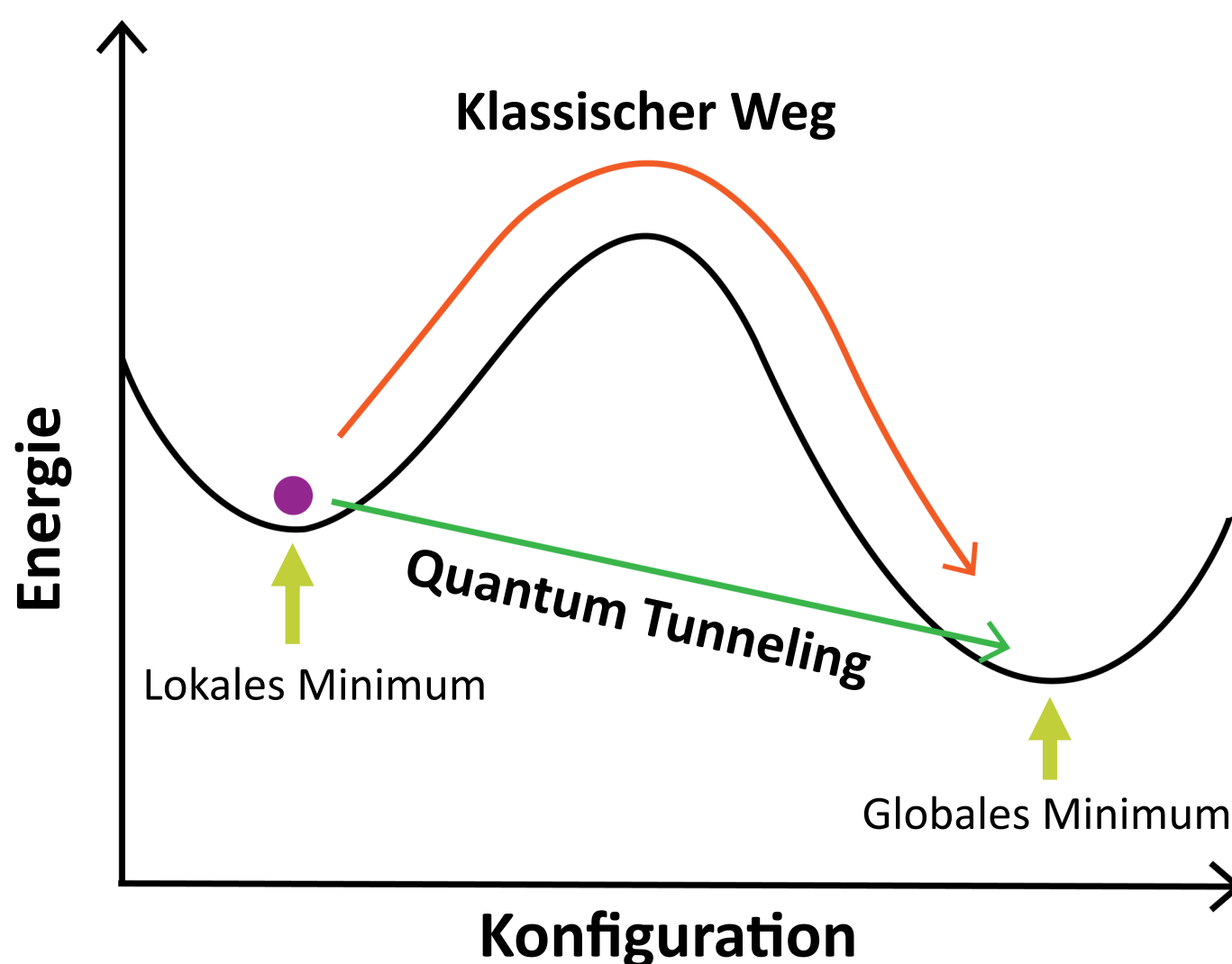


Abb. 6: Tunneleffekte („Quantum Tunneling“) führen vom lokalen zum globalen Minimum.