

Quantum annealing und Logistik

Yanick Kind

12. Juli 2023

Fakultät Physik

Übersicht

Einleitung

Simulated und quantum annealing

Quadratic unconstrained binary optimization

Hardware embedding

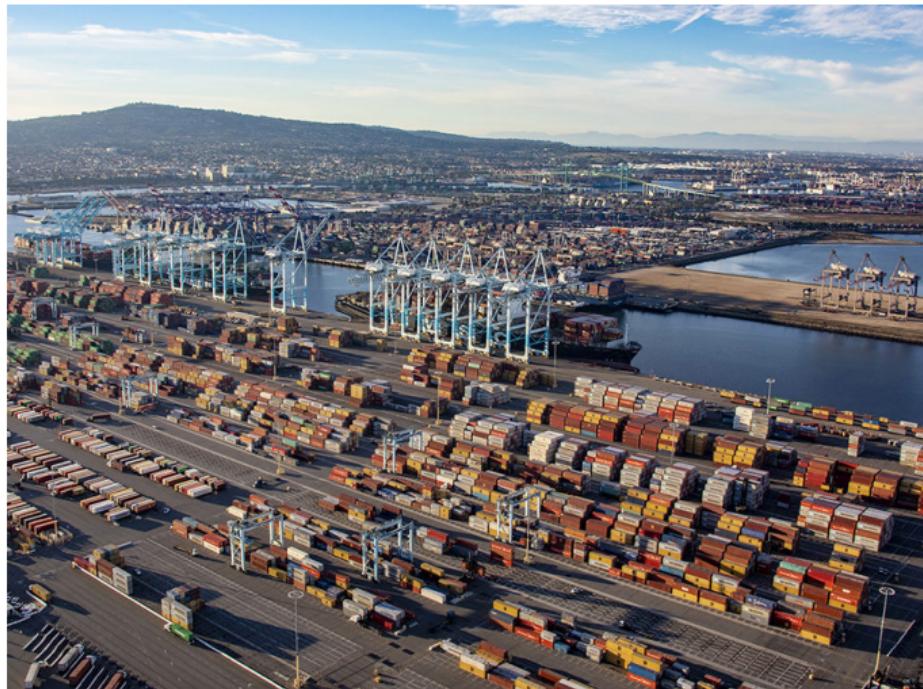
Fallbeispiel Aisin Corporation

Zusammenfassung und Ausblick



azom.com

Motivation



supplychainbrain.com

Traveling Salesman Problem

Traveling Salesman Problem

- klassisches kombinatorische Optimierungsproblem

Traveling Salesman Problem

- klassisches kombinatorische Optimierungsproblem
- Beispiel: Ein Lieferwagen hat n Lieferorte

Traveling Salesman Problem

- klassisches kombinatorische Optimierungsproblem
- Beispiel: Ein Lieferwagen hat n Lieferorte
 - $(n - 1)!$ verschiedene Routen

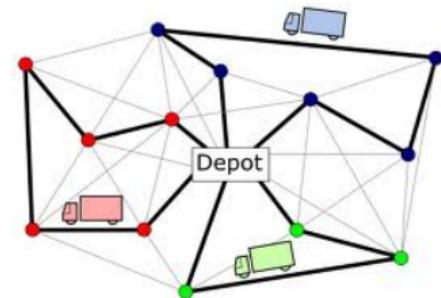
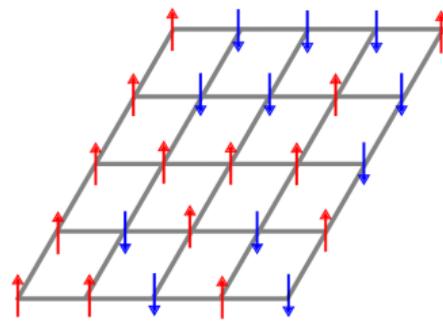
Traveling Salesman Problem

- klassisches kombinatorische Optimierungsproblem
- Beispiel: Ein Lieferwagen hat n Lieferorte
 - $(n - 1)!$ verschiedene Routen
- NP-hard
 - skaliert nicht polynomiell sondern **exponentiell** mit Systemgröße
- Globales Optimum mittels numerischen Lösungsverfahren finden, ist extrem aufwändig

Traveling Salesman Problem

- klassisches kombinatorische Optimierungsproblem
- Beispiel: Ein Lieferwagen hat n Lieferorte
 - $(n - 1)!$ verschiedene Routen
- NP-hard
 - skaliert nicht polynomiell sondern **exponentiell** mit Systemgröße
- Globales Optimum mittels numerischen Lösungsverfahren finden, ist extrem aufwändig
- Local search Algorithmen als heuristische Näherungsverfahren nutzen

Annealing als Lösung?

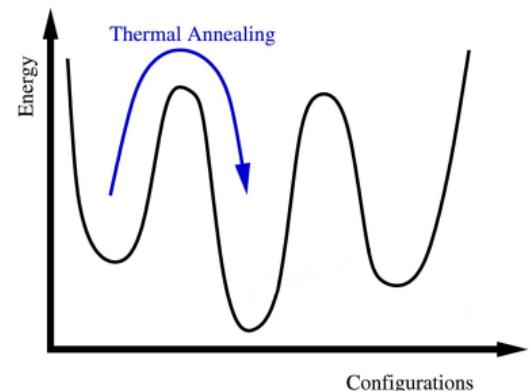


r-bloggers.com

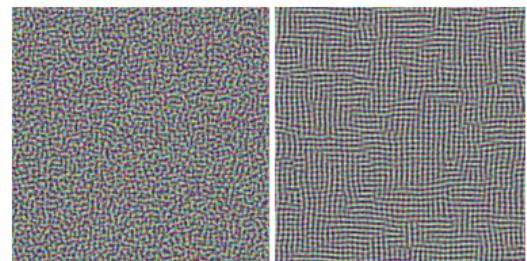
E

f

Simulated annealing (SA)



V. Bapst et al., Physics Reports 523 (2013), editiert

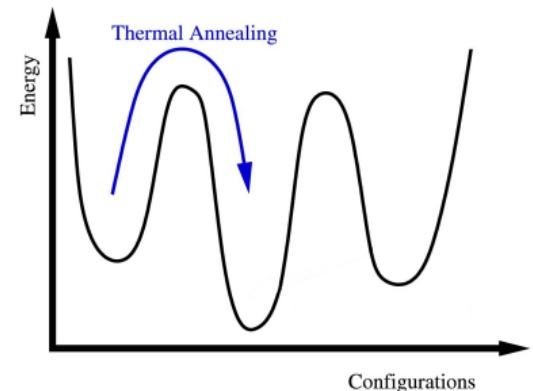


Cyp, wikipedia.org

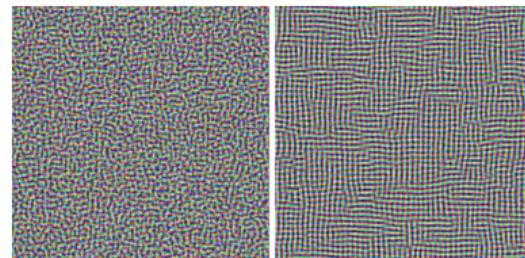
Simulated und quantum annealing

Simulated annealing (SA)

- in Analogie zu auskühlenden Metallen



V. Bapst et al., Physics Reports 523 (2013), editiert

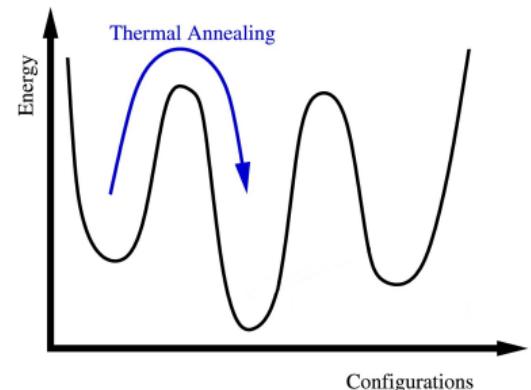


Cyp, wikipedia.org

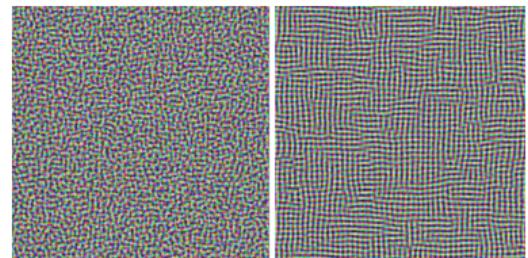
Simulated und quantum annealing

Simulated annealing (SA)

- in Analogie zu auskühlenden Metallen
 - schneller auskühlen → suboptimaler Zustand (amorph)



V. Bapst et al., Physics Reports 523 (2013), editiert

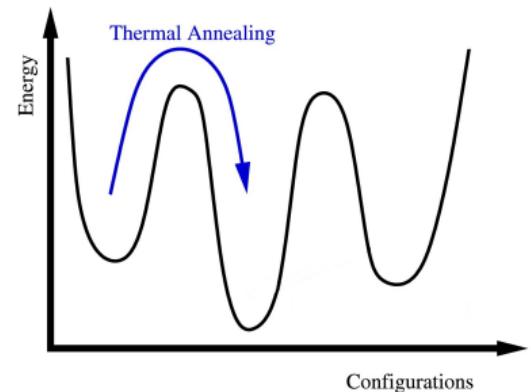


Cyp, wikipedia.org

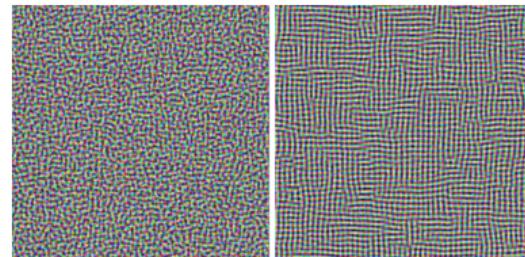
Simulated und quantum annealing

Simulated annealing (SA)

- in Analogie zu auskühlenden Metallen
 - schneller auskühlen → suboptimaler Zustand (amorph)
 - langsamer auskühlen → optimalerer Zustand (kristallin)



V. Bapst et al., Physics Reports 523 (2013), editiert

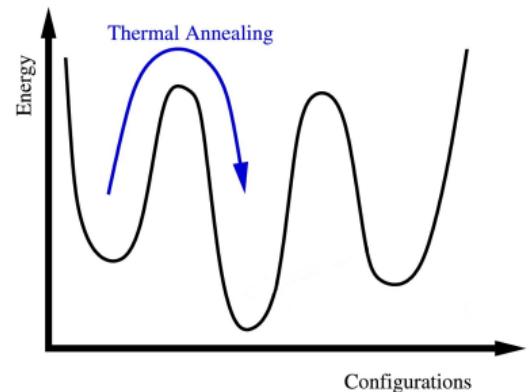


Cyp, wikipedia.org

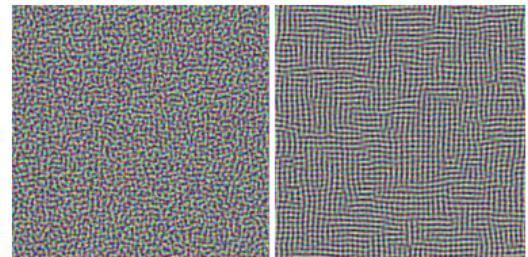
Simulated und quantum annealing

Simulated annealing (SA)

- in Analogie zu auskühlenden Metallen
 - schneller auskühlen → suboptimaler Zustand (amorph)
 - langsamer auskühlen → optimalerer Zustand (kristallin)
- nutzt thermische Fluktuationen, um lokalen Minima zu entkommen



V. Bapst et al., Physics Reports 523 (2013), editiert

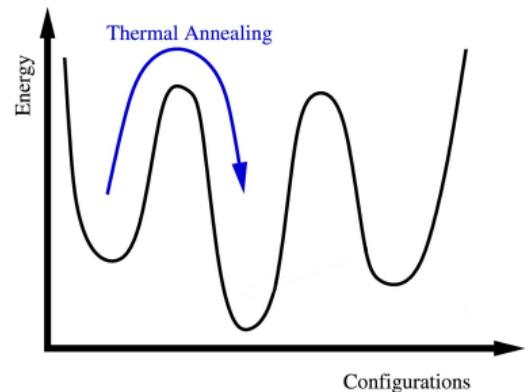


Cyp, wikipedia.org

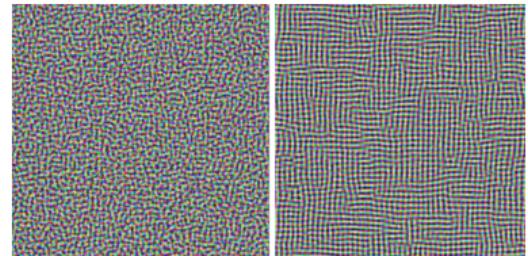
Simulated und quantum annealing

Simulated annealing (SA)

- in Analogie zu auskühlenden Metallen
 - schneller auskühlen → suboptimaler Zustand (amorph)
 - langsamer auskühlen → optimalerer Zustand (kristallin)
- nutzt thermische Fluktuationen, um lokalen Minima zu entkommen
- Wahrscheinlichkeit P_i folgen Mastergleichung $\frac{dP_i}{dt} = \sum_j L_{ij} P_j$



V. Bapst et al., Physics Reports 523 (2013), editiert



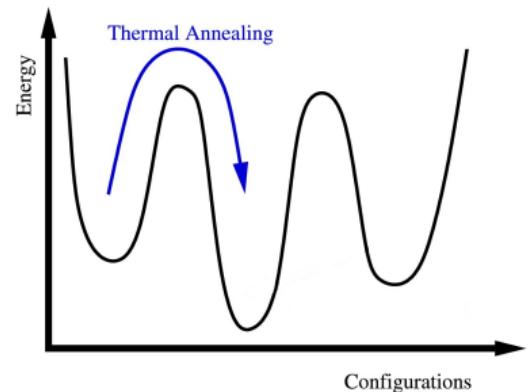
Cyp, wikipedia.org

Simulated und quantum annealing

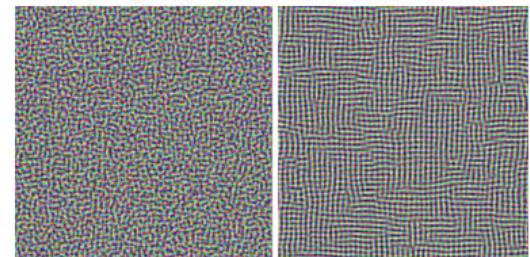
Simulated annealing (SA)

- in Analogie zu auskühlenden Metallen
 - schneller auskühlen → suboptimaler Zustand (amorph)
 - langsamer auskühlen → optimalerer Zustand (kristallin)
- nutzt thermische Fluktuationen, um lokalen Minima zu entkommen
- Wahrscheinlichkeit P_i folgen Mastergleichung $\frac{dP_i}{dt} = \sum_j L_{ij} P_j$

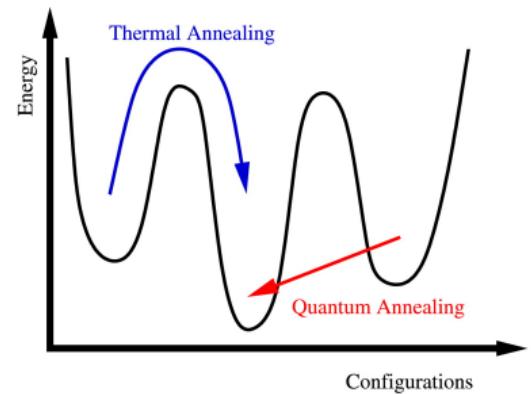
$$L_{ij} = \begin{cases} \left[1 + \exp((E_i - E_j)/T_t)\right]^{-1} & \text{einzelne Spin-Differenz} \\ -\sum_{k \neq i} L_{ki} & i = j \\ 0 & \text{sonst} \end{cases}$$



V. Bapst et al., Physics Reports 523 (2013), editiert



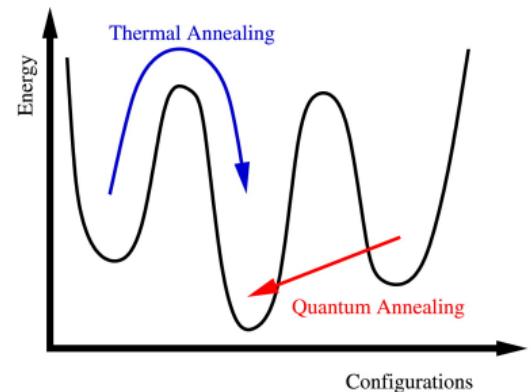
Quantum Annealing (QA)



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

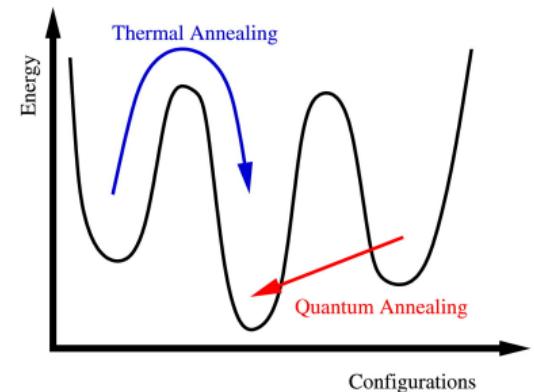
- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^z}_{H_f} - \underbrace{\Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

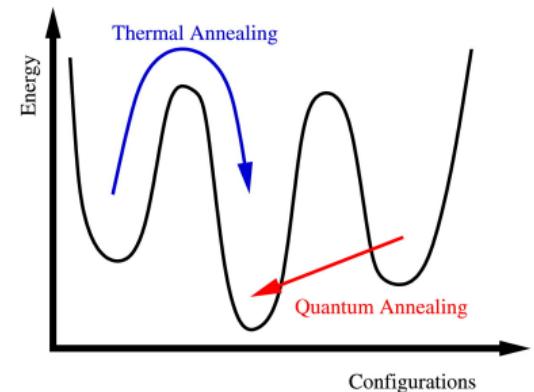
- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \sum_i \sigma_i^z - \Gamma(t) \sum_i \sigma_i^x$
- $H_x(t)$ verursacht Tunneln zwischen Eigenzuständen von H_0



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

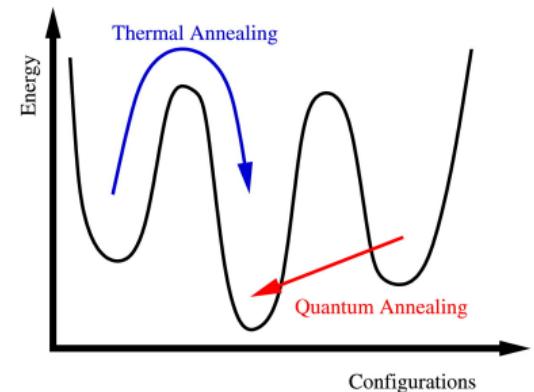
- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \sum_i \sigma_i^z - \Gamma(t) \sum_i \sigma_i^x$
- $H_x(t)$ verursacht Tunneln zwischen Eigenzuständen von H_0



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

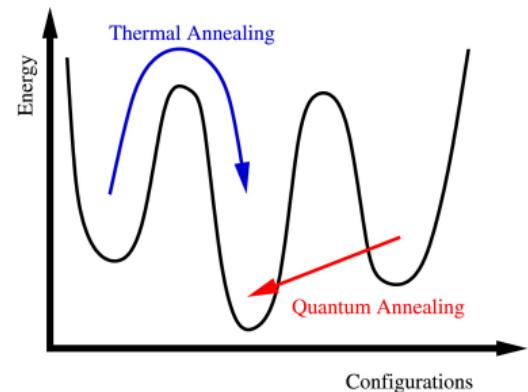
- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^z}_{H_f} - \underbrace{\Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$
- $H_x(t)$ verursacht Tunneln zwischen Eigenzuständen von H_0
 - schnellere Konvergenz



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

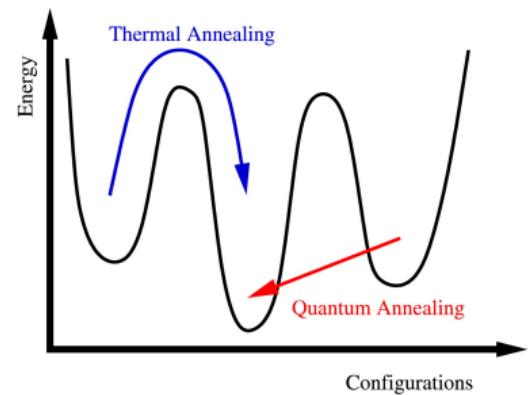
- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^z}_{H_f} - \underbrace{\Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$
- $H_x(t)$ verursacht Tunneln zwischen Eigenzuständen von H_0
 - schnellere Konvergenz
- $|\Psi(t)\rangle$ durch Schrödinger-Gleichung $i \frac{\partial |\Psi(t)\rangle}{\partial t} = H(t) |\Psi(t)\rangle$ festgelegt



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

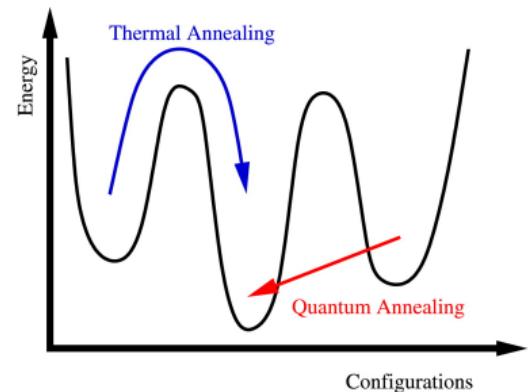
- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^z}_{H_f} - \Gamma(t) \sum_i \sigma_i^x$
- $H_x(t)$ verursacht Tunneln zwischen Eigenzuständen von H_0
 - schnellere Konvergenz
- $|\Psi(t)\rangle$ durch Schrödinger-Gleichung $i \frac{\partial |\Psi(t)\rangle}{\partial t} = H(t) |\Psi(t)\rangle$ festgelegt
- wähle monoton fallende Funktionen für $\Gamma(t)$ („ausglühen“)



V. Bapst et al., Physics Reports 523 (2013)

Quantum Annealing (QA)

- transversales Ising-Modell $H(t) = \underbrace{-\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^z}_{H_f} - \Gamma(t) \sum_i \sigma_i^x$
- $H_x(t)$ verursacht Tunneln zwischen Eigenzuständen von H_0
 - schnellere Konvergenz
- $|\Psi(t)\rangle$ durch Schrödingergleichung $i \frac{\partial |\Psi(t)\rangle}{\partial t} = H(t) |\Psi(t)\rangle$ festgelegt
- wähle monoton fallende Funktionen für $\Gamma(t)$ („ausglühen“)

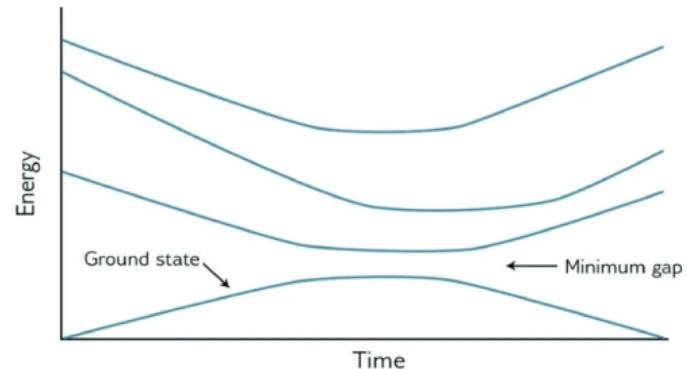


V. Bapst et al., Physics Reports 523 (2013)

Adiabatisches Theorem der Quantenmechanik

System bleibt im selben Eigenzustand von H , wenn sich H hinreichend langsam ändert

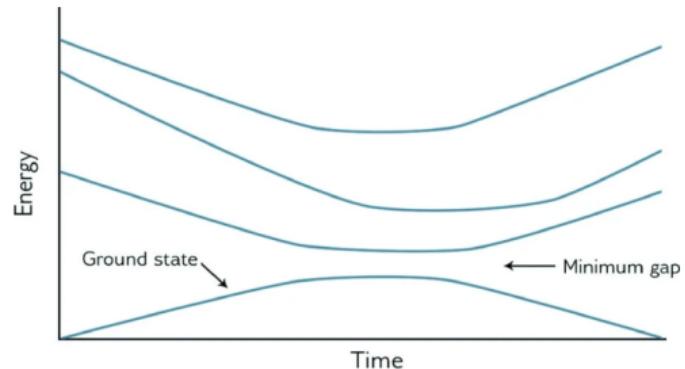
Adiabatische Zeit



E. R. Miranda (Ed.), Quantum Computer Music (2022)

Adiabatische Zeit

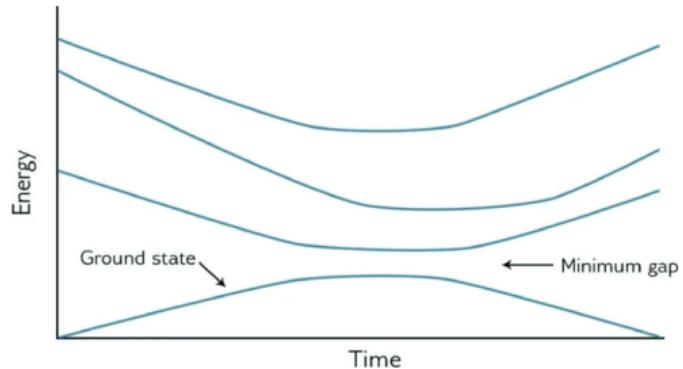
- minimale Zeit, damit sich System adiabatisch ändert τ



E. R. Miranda (Ed.), Quantum Computer Music (2022)

Adiabatische Zeit

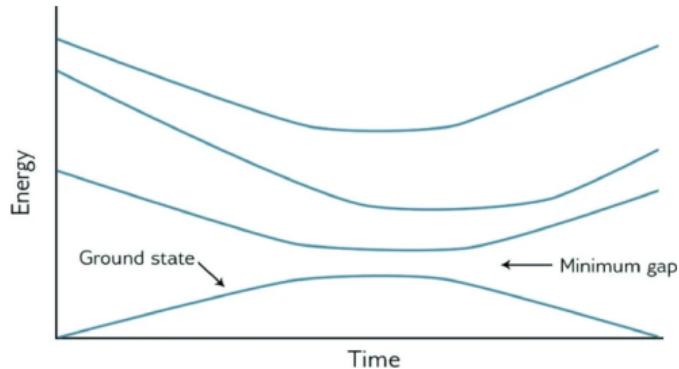
- minimale Zeit, damit sich System adiabatisch ändert τ
- $\Delta(s) = E_1(s) - E_0(s)$



E. R. Miranda (Ed.), Quantum Computer Music (2022)

Adiabatische Zeit

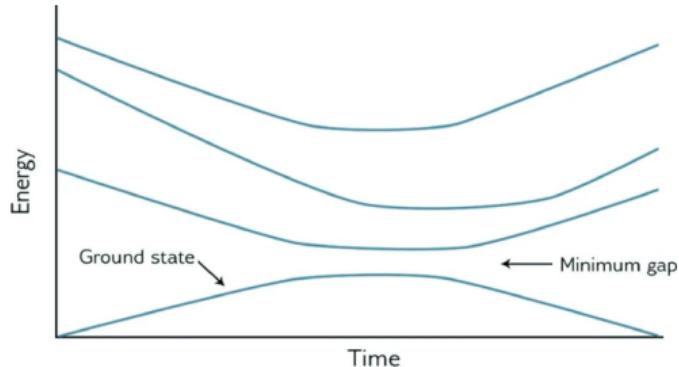
- minimale Zeit, damit sich System adiabatisch ändert τ
- $\Delta(s) = E_1(s) - E_0(s)$
- mindestens $\tau \propto \frac{1}{\Delta(s)^2}$, schlimmstenfalls $\tau \propto \frac{1}{\Delta(s)^3}$



E. R. Miranda (Ed.), Quantum Computer Music (2022)

Adiabatische Zeit

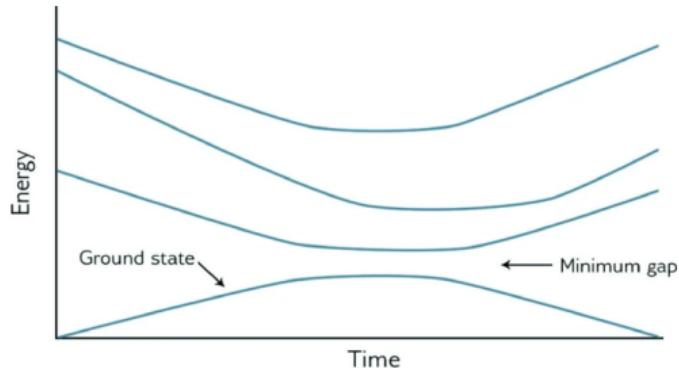
- minimale Zeit, damit sich System adiabatisch ändert τ
- $\Delta(s) = E_1(s) - E_0(s)$
- mindestens $\tau \propto \frac{1}{\Delta(s)^2}$, schlimmstenfalls $\tau \propto \frac{1}{\Delta(s)^3}$
- Problem: $\Delta(s)$ sehr klein



E. R. Miranda (Ed.), Quantum Computer Music (2022)

Adiabatische Zeit

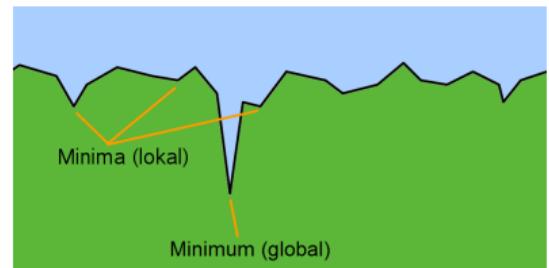
- minimale Zeit, damit sich System adiabatisch ändert τ
- $\Delta(s) = E_1(s) - E_0(s)$
- mindestens $\tau \propto \frac{1}{\Delta(s)^2}$, schlimmstenfalls $\tau \propto \frac{1}{\Delta(s)^3}$
- Problem: $\Delta(s)$ sehr klein
 - $\tau \rightarrow \infty$



E. R. Miranda (Ed.), Quantum Computer Music (2022)

Simulation zu simulated und quantum annealing

$$H(t) = \underbrace{- \sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \sum_i \sigma_i^z - \Gamma(t) \sum_i \sigma_i^x$$

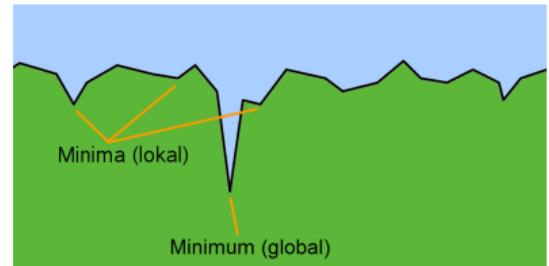


Fleischer, K., wikipedia.org

Simulation zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H
- $\lim_{t \rightarrow 0} \Gamma(t) = \lim_{t \rightarrow 0} T(t) = \infty$

$$H(t) = - \underbrace{\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \underbrace{\sum_i \sigma_i^z}_{\Gamma(t)} \underbrace{- \Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$$

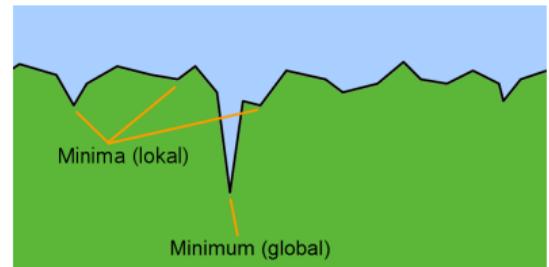


Fleischer, K., wikipedia.org

Simulation zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H
- $\lim_{t \rightarrow 0} \Gamma(t) = \lim_{t \rightarrow 0} T(t) = \infty$
 - Superposition aller Zustände gleicher Amplitude (QA)
 - Alle Zustände gleich wahrscheinlich (SA)

$$H(t) = - \underbrace{\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \underbrace{\sum_i \sigma_i^z}_{\Gamma(t)} \underbrace{- \Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$$

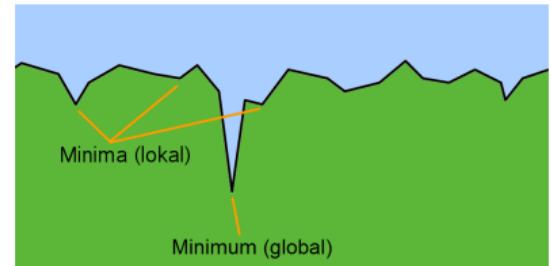


Fleischer, K., wikipedia.org

Simulation zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H
- $\lim_{t \rightarrow 0} \Gamma(t) = \lim_{t \rightarrow 0} T(t) = \infty$
 - Superposition aller Zustände gleicher Amplitude (QA)
 - Alle Zustände gleich wahrscheinlich (SA)
- Idealfall (adiabatisch): $\lim_{t \rightarrow \infty} P_{QA}(t) = \lim_{t \rightarrow \infty} P_{SA}(t) = 1$

$$H(t) = - \underbrace{\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \underbrace{\sum_i \sigma_i^z}_{\Gamma(t)} \underbrace{- \Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$$

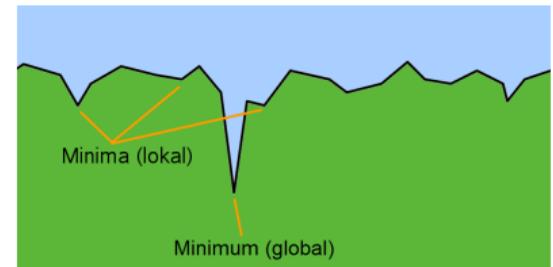


Fleischer, K., wikipedia.org

Simulation zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H
- $\lim_{t \rightarrow 0} \Gamma(t) = \lim_{t \rightarrow 0} T(t) = \infty$
 - Superposition aller Zustände gleicher Amplitude (QA)
 - Alle Zustände gleich wahrscheinlich (SA)
- Idealfall (adiabatisch): $\lim_{t \rightarrow \infty} P_{QA}(t) = \lim_{t \rightarrow \infty} P_{SA}(t) = 1$
 - Grundzustand gefunden

$$H(t) = - \underbrace{\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \underbrace{\sum_i \sigma_i^z}_{\Gamma(t)} \underbrace{- \Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$$

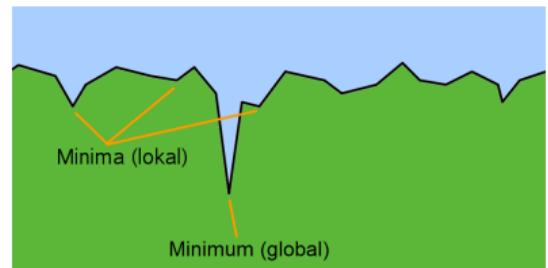


Fleischer, K., wikipedia.org

Simulation zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H
- $\lim_{t \rightarrow 0} \Gamma(t) = \lim_{t \rightarrow 0} T(t) = \infty$
 - Superposition aller Zustände gleicher Amplitude (QA)
 - Alle Zustände gleich wahrscheinlich (SA)
- Idealfall (adiabatisch): $\lim_{t \rightarrow \infty} P_{QA}(t) = \lim_{t \rightarrow \infty} P_{SA}(t) = 1$
 - Grundzustand gefunden
 - sonst: In einem lokalen Minimum „hängen“ geblieben

$$H(t) = - \underbrace{\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \underbrace{\sum_i \sigma_i^z}_{\Gamma(t)} \underbrace{- \Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$$

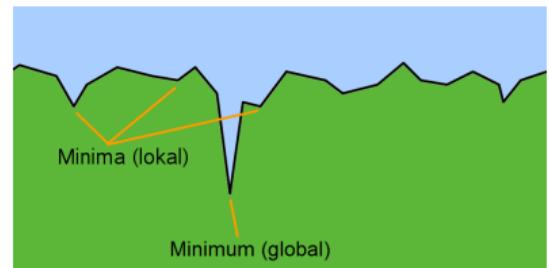


Fleischer, K., wikipedia.org

Simulation zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H
- $\lim_{t \rightarrow 0} \Gamma(t) = \lim_{t \rightarrow 0} T(t) = \infty$
 - Superposition aller Zustände gleicher Amplitude (QA)
 - Alle Zustände gleich wahrscheinlich (SA)
- Idealfall (adiabatisch): $\lim_{t \rightarrow \infty} P_{QA}(t) = \lim_{t \rightarrow \infty} P_{SA}(t) = 1$
 - Grundzustand gefunden
 - sonst: In einem lokalen Minimum „hängen“ geblieben
- 8 Spins

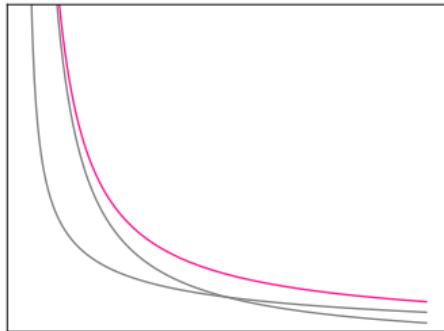
$$H(t) = - \underbrace{\sum_{ij} J_{ij} \sigma_i^z \sigma_j^z}_{H_f} - h \underbrace{\sum_i \sigma_i^z}_{\Gamma(t)} \underbrace{- \Gamma(t) \sum_i \sigma_i^x}_{H_i(t)}$$



Fleischer, K., wikipedia.org

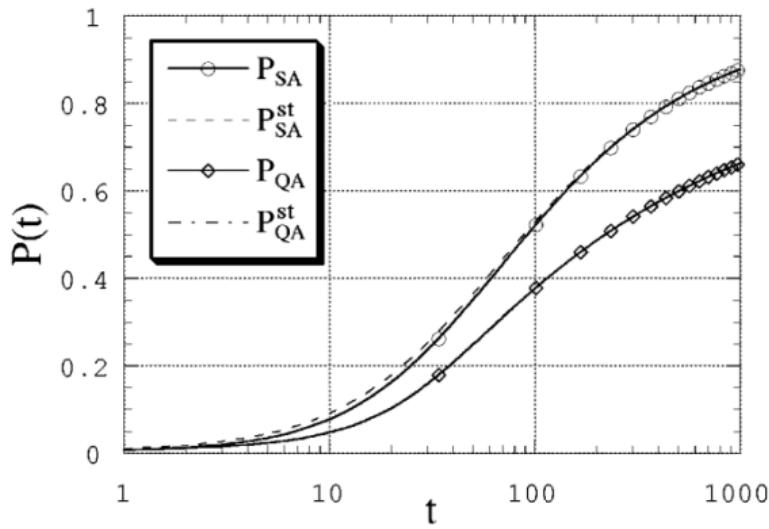
Ergebnisse zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H



■ $J_{ij} = const > 0$
■ $h = 0.1$

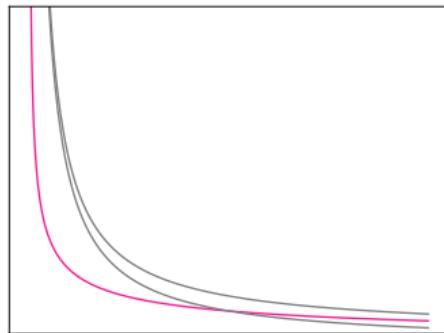
$$\Gamma(t) = T(t) = 3/\ln(t + 1)$$



Kadowaki, T., Nishimori, H., PHYSICAL REVIEW E 1998

Ergebnisse zu simulated und quantum annealing

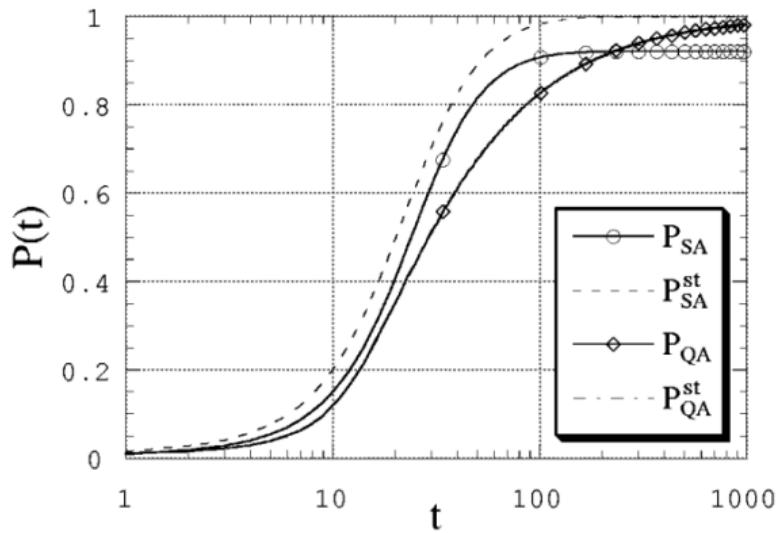
- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H



Kadowaki, T., Nishimori, H., PHYSICAL REVIEW E 1998

- $J_{ij} = const > 0$
- $h = 0.1$

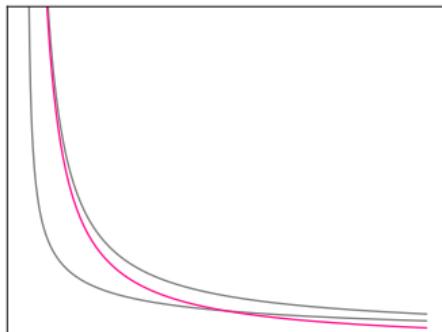
$$\Gamma(t) = T(t) = 3/\sqrt{t}$$



Simulated und quantum annealing

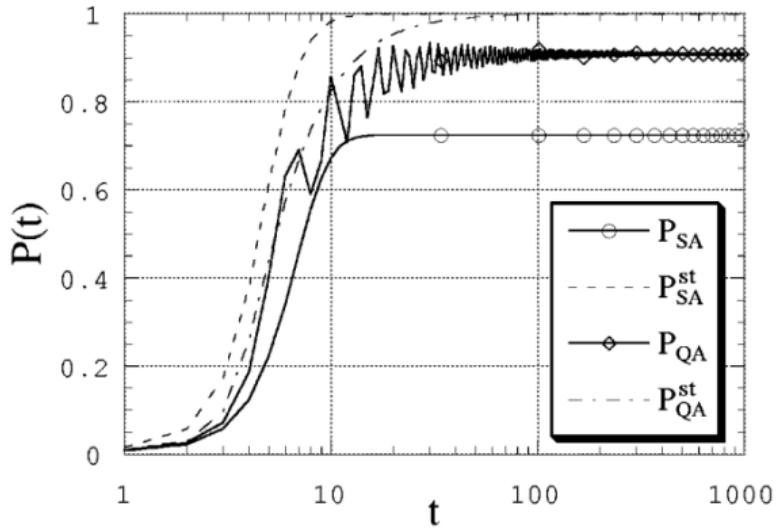
Ergebnisse zu simulated und quantum annealing

- $P_{SA}(t)$: Wahrscheinlichkeit System in GZ zu finden
- $P_{QA}(t) = |\langle g | \Psi(t) \rangle|^2$: Wahrscheinlichkeit System in GZ $|g\rangle$ von H_f zu finden
- $P_{SA}^{st}(T)$: Boltzmann-Faktor des GZ von H_f
- $P_{QA}^{st}(\Gamma) = |\langle g | \Psi_\Gamma \rangle|^2$ mit $|\Psi_\Gamma\rangle$ als GZ von H



■ $J_{ij} = const > 0$
■ $h = 0.1$

$$\Gamma(t) = T(t) = 3/t$$



Kadowaki, T., Nishimori, H., PHYSICAL REVIEW E 1998

Binäre Variable

Binäre Variable

- $x \in \{0, 1\}$

Binäre Variable

- $x \in \{0, 1\}$
- Beispiel: Aufenthaltsort von einem Truck
- Formulierung als Graph mit N Knoten i

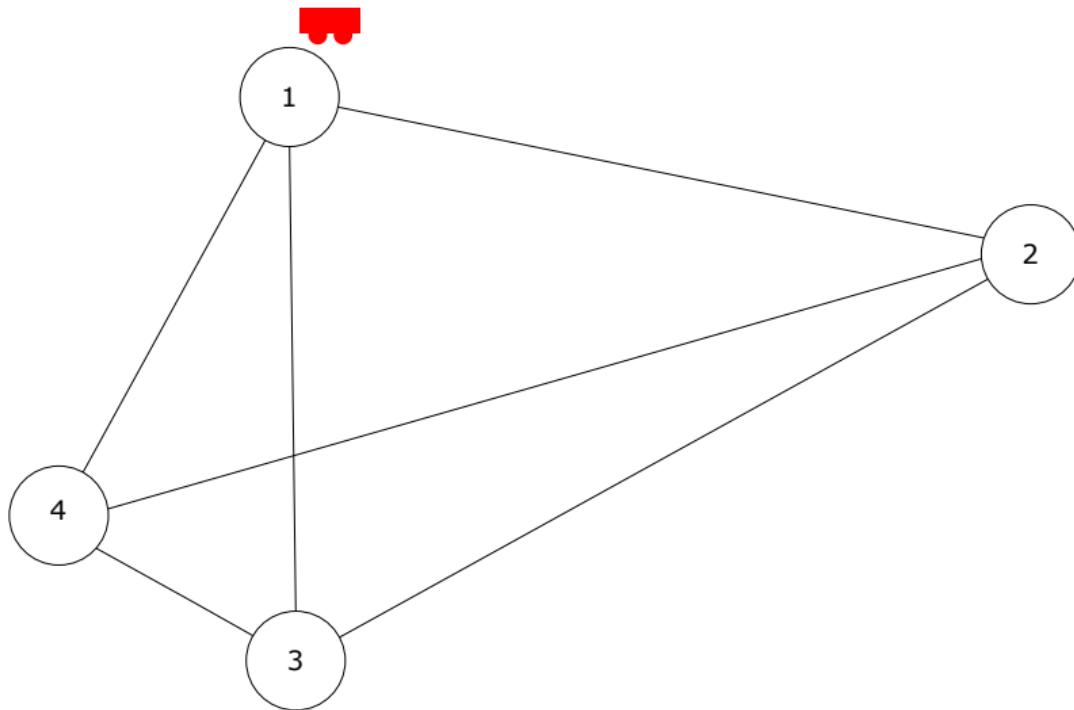
Binäre Variable

- $x \in \{0, 1\}$
- Beispiel: Aufenthaltsort von einem Truck
- Formulierung als Graph mit N Knoten i
 - ordne jedem Knoten i eine binäre Variable zu

Binäre Variable

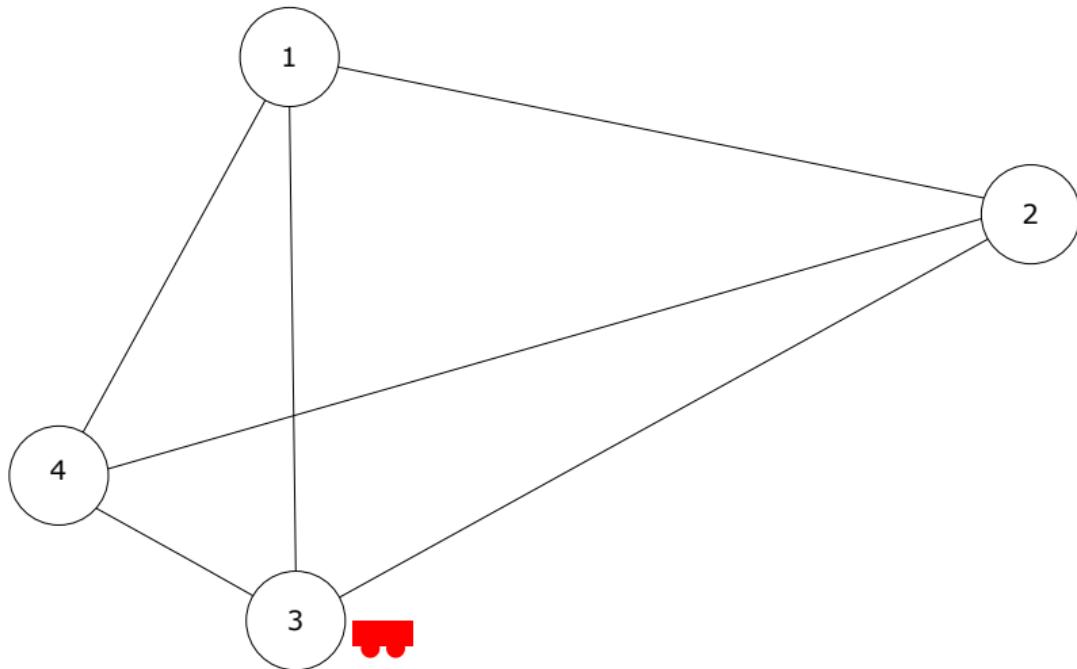
- $x \in \{0, 1\}$
- Beispiel: Aufenthaltsort von einem Truck
- Formulierung als Graph mit N Knoten i
 - ordne jedem Knoten i eine binäre Variable zu
 - jedem Truck kann ein Vektor $x^j = (x_1^j, \dots, x_N^j)$ von binären Variablen zugeordnet werden
- Truck j an Knoten $i \rightarrow x_i^j = 1$ sonst $x_k^j = 0$

Binäre Variable



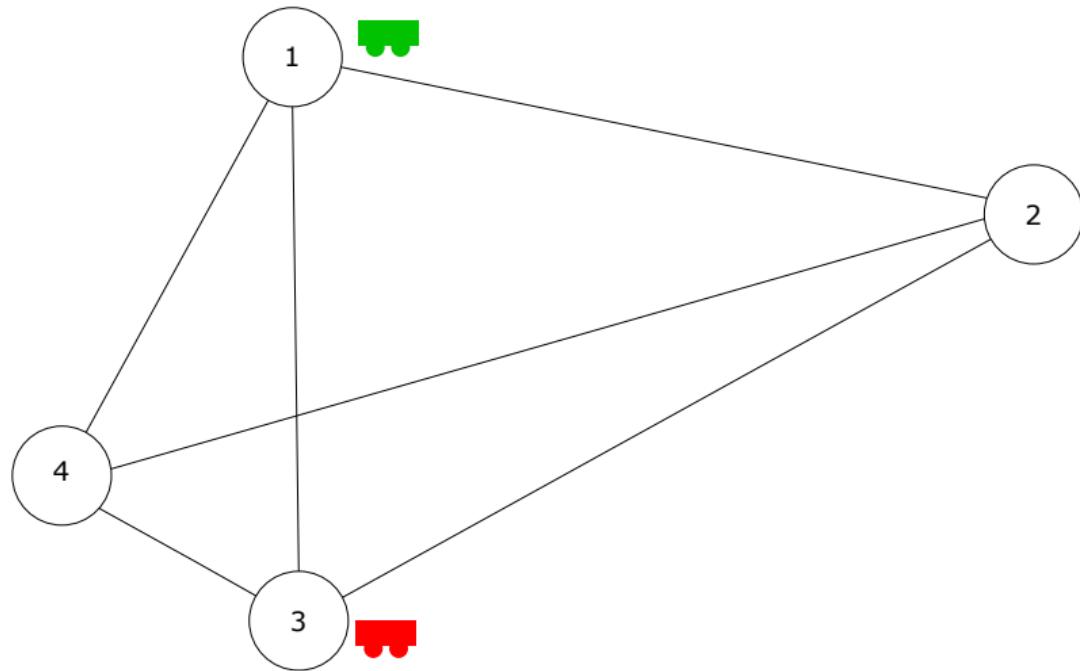
$$\underline{x}^{\text{rot}} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Binäre Variable



$$\underline{x}^{\text{rot}} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

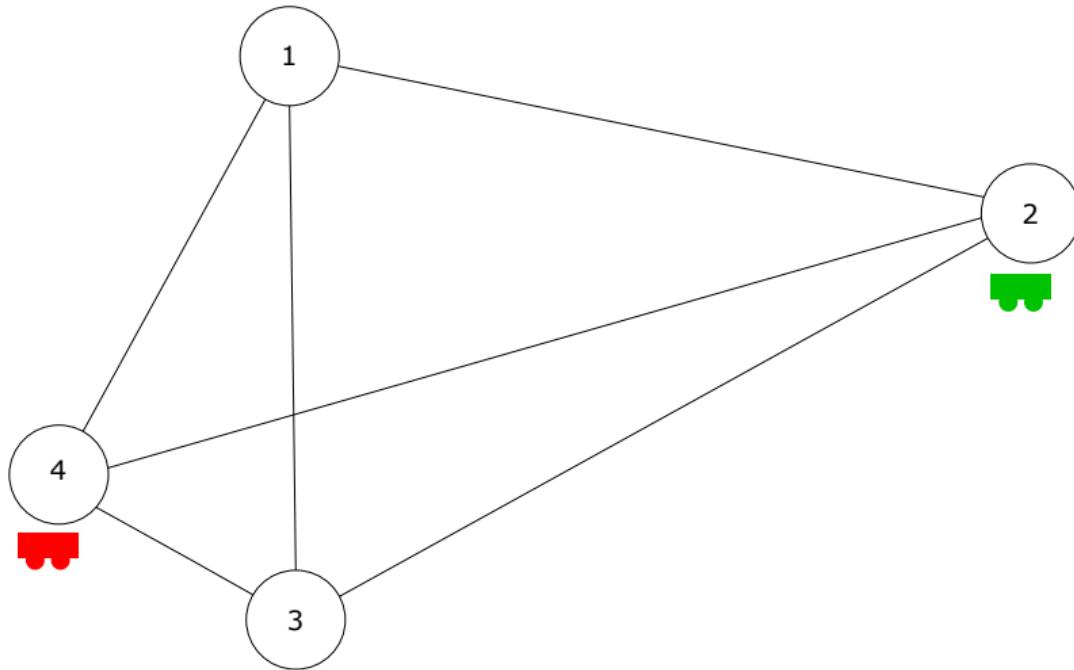
Binäre Variable



$$\underline{x}^{\text{rot}} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\underline{x}^{\text{grün}} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Binäre Variable



$$\underline{x}^{\text{rot}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

$$\underline{x}^{\text{grün}} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

Quadratic unconstrained binary optimization (QUBO)

Quadratic unconstrained binary optimization (QUBO)

Aufgabe

$$\text{Minimiere } f(\underline{x}, \underline{\underline{Q}}) = \sum_{ij} Q_{ij} x_i x_j$$

Quadratic unconstrained binary optimization (QUBO)

Aufgabe

$$\text{Minimiere } f(\underline{x}, \underline{\underline{Q}}) = \sum_{ij} Q_{ij} x_i x_j$$

- $x_i \in \{0, 1\}$ als binäre Variable

Quadratic unconstrained binary optimization (QUBO)

Aufgabe

$$\text{Minimiere } f(\underline{x}, \underline{\underline{Q}}) = \sum_{ij} Q_{ij} x_i x_j$$

- $x_i \in \{0, 1\}$ als binäre Variable
 - lässt sich mittels $s_i = 2x_i - 1 \in \{-1, 1\}$ auf Ising-Variablen („Spin-1/2“) transformieren

Quadratic unconstrained binary optimization (QUBO)

Aufgabe

$$\text{Minimiere } f(\underline{x}, \underline{\underline{Q}}) = \sum_{ij} Q_{ij} x_i x_j$$

- $x_i \in \{0, 1\}$ als binäre Variable
 - lässt sich mittels $s_i = 2x_i - 1 \in \{-1, 1\}$ auf Ising-Variablen („Spin-1/2“) transformieren
- Q_{ij} definiert Wechselwirkung zwischen x_i

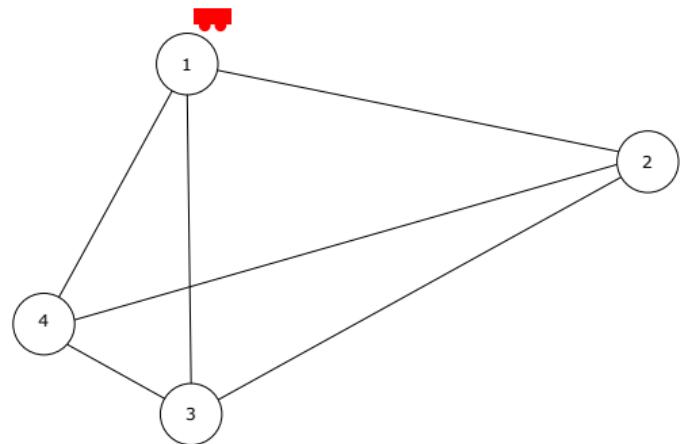
Quadratic unconstrained binary optimization (QUBO)

Aufgabe

$$\text{Minimiere } f(\underline{x}, \underline{\underline{Q}}) = \sum_{ij} Q_{ij} x_i x_j$$

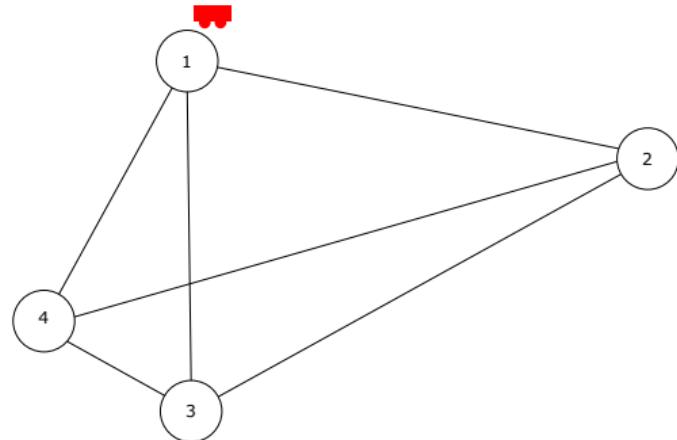
- $x_i \in \{0, 1\}$ als binäre Variable
 - lässt sich mittels $s_i = 2x_i - 1 \in \{-1, 1\}$ auf Ising-Variablen („Spin-1/2“) transformieren
- Q_{ij} definiert Wechselwirkung zwischen x_i
 - allg. hermitesch (symmetrisch im Ising-Modell)

Quadratic unconstrained binary optimization constraints



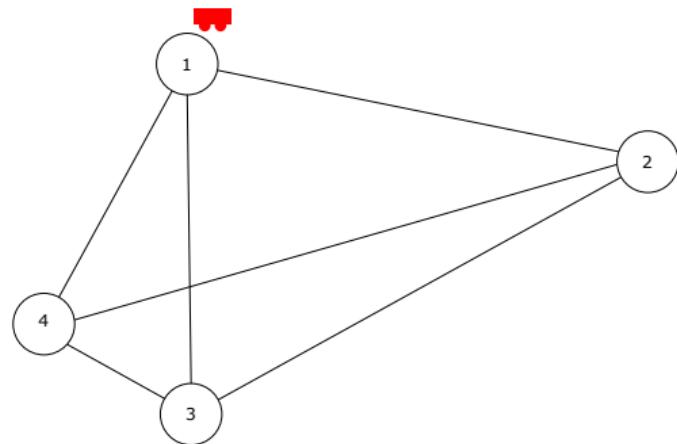
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren



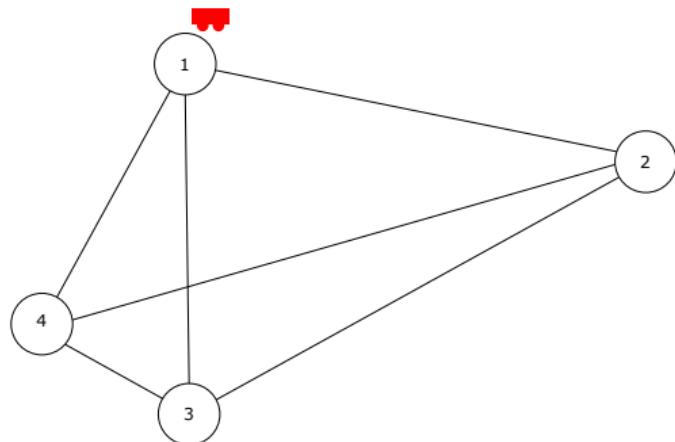
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren



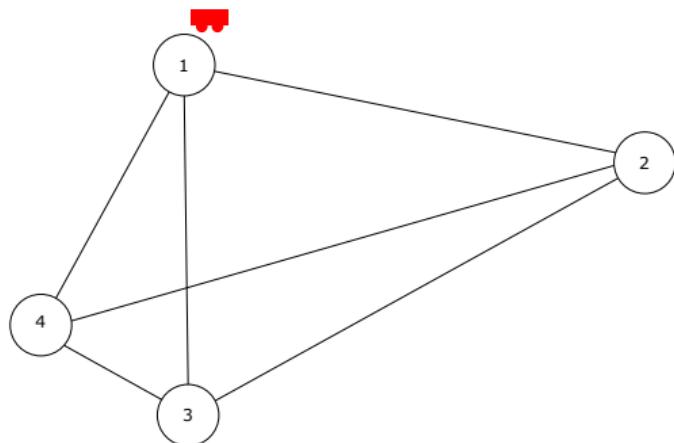
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren
- Beispiel: Ein Truck darf sich nur an einem Punkt aufhalten



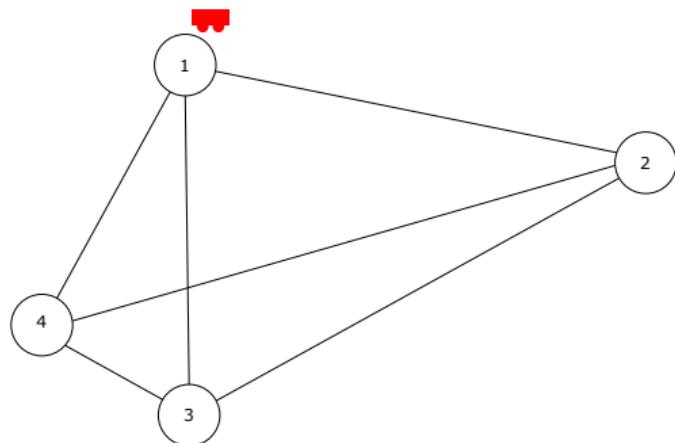
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren
- Beispiel: Ein Truck darf sich nur an einem Punkt aufhalten
 - $\sum_i x_i = 1 \iff \sum_i x_i - 1 = 0$



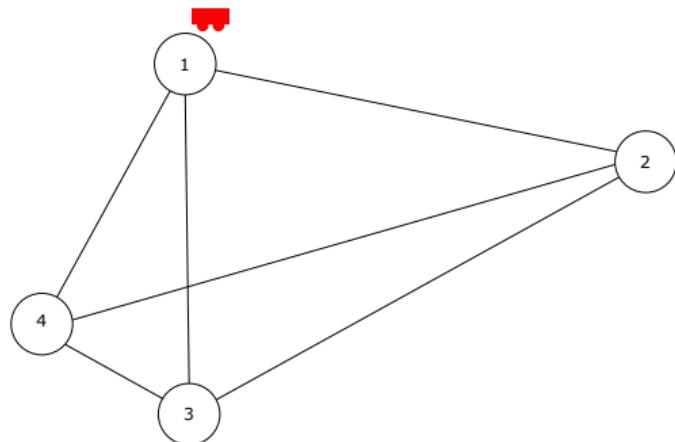
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren
- Beispiel: Ein Truck darf sich nur an einem Punkt aufhalten
 - $\sum_i x_i = 1 \iff \sum_i x_i - 1 = 0$
 - in QUBO Problem umwandeln $\rightarrow g(x_i) = (\sum_i x_i - 1)^2$



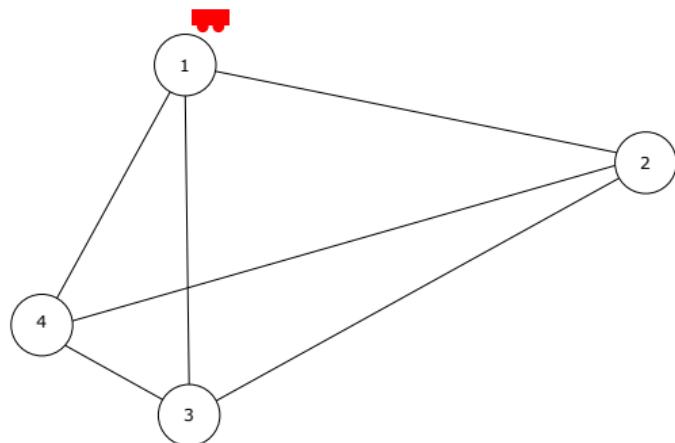
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren
- Beispiel: Ein Truck darf sich nur an einem Punkt aufhalten
 - $\sum_i x_i = 1 \iff \sum_i x_i - 1 = 0$
 - in QUBO Problem umwandeln $\rightarrow g(x_i) = (\sum_i x_i - 1)^2$
 - $f \rightarrow f + \lambda g$



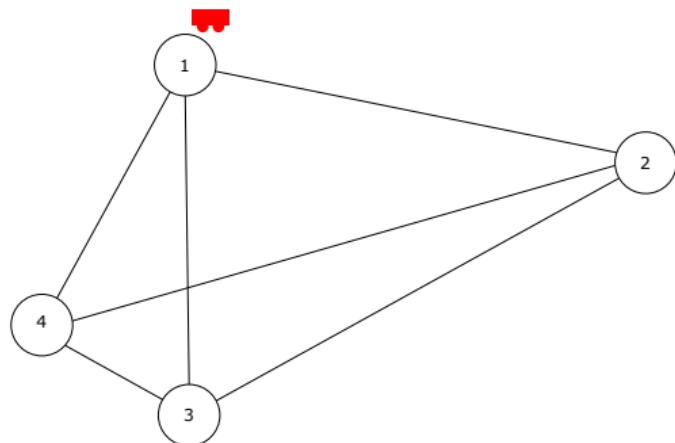
Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren
- Beispiel: Ein Truck darf sich nur an einem Punkt aufhalten
 - $\sum_i x_i = 1 \iff \sum_i x_i - 1 = 0$
 - in QUBO Problem umwandeln $\rightarrow g(x_i) = (\sum_i x_i - 1)^2$
 - $f \rightarrow f + \lambda g$
 - constraint verletzt → Kostenfunktion erhöht



Quadratic unconstrained binary optimization constraints

- constraints lassen sich nicht direkt in der Hardware implementieren
 - constraints als „Strafe“ in der Kostenfunktion implementieren
- Beispiel: Ein Truck darf sich nur an einem Punkt aufhalten
 - $\sum_i x_i = 1 \iff \sum_i x_i - 1 = 0$
 - in QUBO Problem umwandeln $\rightarrow g(x_i) = (\sum_i x_i - 1)^2$
 - $f \rightarrow f + \lambda g$
 - constraint verletzt → Kostenfunktion erhöht
- Lagrangeparameter λ sehr groß wählen
 - constraint eingehalten

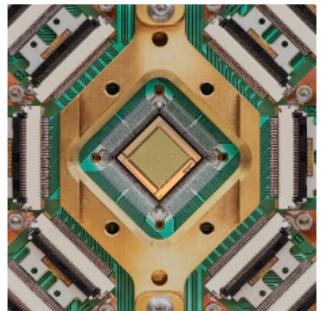


Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer



techcrunch.com



dwavesys.com

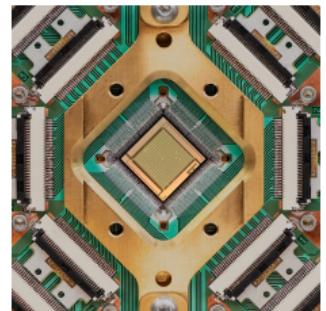
Hardware embedding

Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer
- Quantum Processing Unit (QPU) mit qubits und coupler



techcrunch.com



dwavesys.com

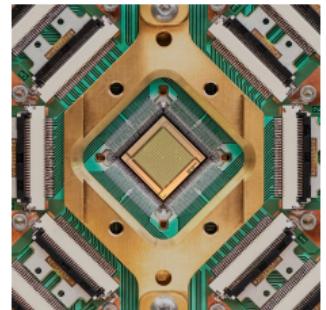
Hardware embedding

Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer
- Quantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet



techcrunch.com



dwavesys.com

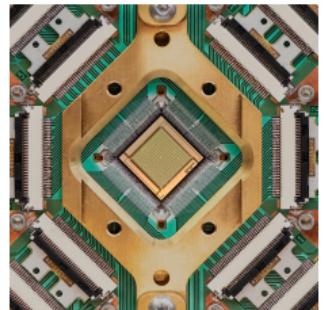
Hardware embedding

Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit



techcrunch.com



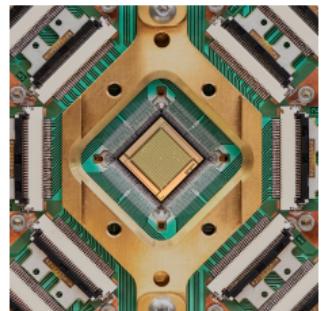
dwavesys.com

Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen



techcrunch.com



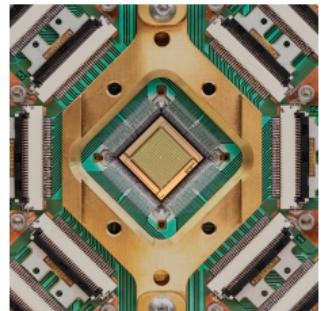
dwavesys.com

Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen
 - mögliches Energieminimum gefunden



techcrunch.com



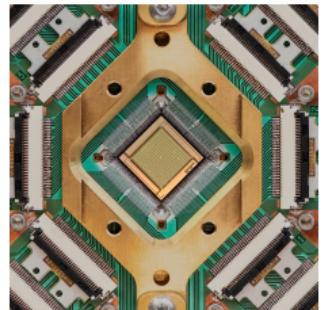
dwavesys.com

Quantum annealer von D-Wave Systems

- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen
 - mögliches Energieminimum gefunden



techcrunch.com



dwavesys.com

Unterschied zu Quantencomputer

Quantum annealer von D-Wave Systems

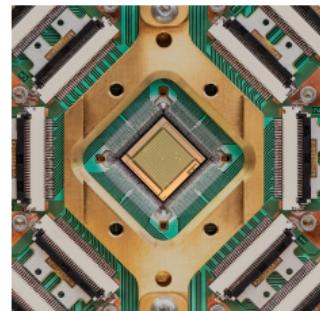
- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen
 - mögliches Energieminimum gefunden



techcrunch.com

Unterschied zu Quantencomputer

- Quantum annealer nur für kombinatorische Optimierungsprobleme



dwavesys.com

Quantum annealer von D-Wave Systems

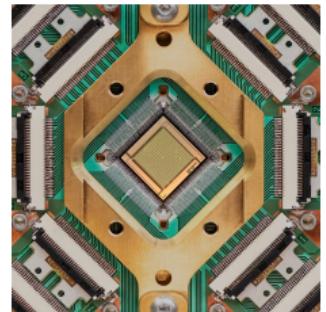
- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen
 - mögliches Energieminimum gefunden



techcrunch.com

Unterschied zu Quantencomputer

- Quantum annealer nur für kombinatorische Optimierungsprobleme
- Quantencomputer nutzen Quantengatter (unitäre Operationen)



dwavesys.com

Quantum annealer von D-Wave Systems

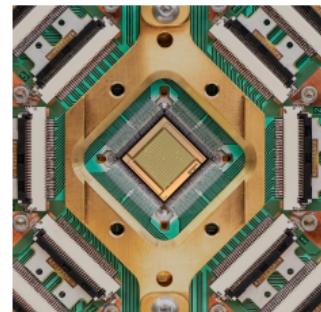
- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen
 - mögliches Energieminimum gefunden



techcrunch.com

Unterschied zu Quantencomputer

- Quantum annealer nur für kombinatorische Optimierungsprobleme
- Quantencomputer nutzen Quantengatter (unitäre Operationen)
 - kann mehr Probleme lösen



dwavesys.com

Quantum annealer von D-Wave Systems

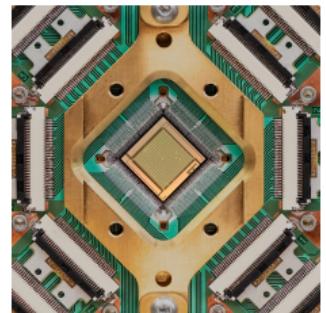
- D-Wave Systems als Hersteller von quantum annealer
- Qantum Processing Unit (QPU) mit qubits und coupler
 - qubits und coupler sind gewichtet
- neuste Version: 5000 qubits mit 15 coupler pro qubit
- Nach annealing-Prozess die qubits auslesen
 - mögliches Energieminimum gefunden



techcrunch.com

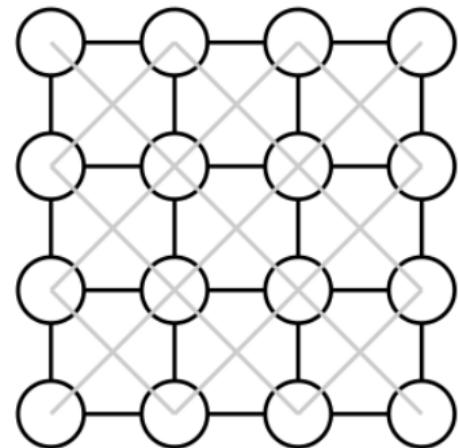
Unterschied zu Quantencomputer

- Quantum annealer nur für kombinatorische Optimierungsprobleme
- Quantencomputer nutzen Quantengatter (unitäre Operationen)
 - kann mehr Probleme lösen



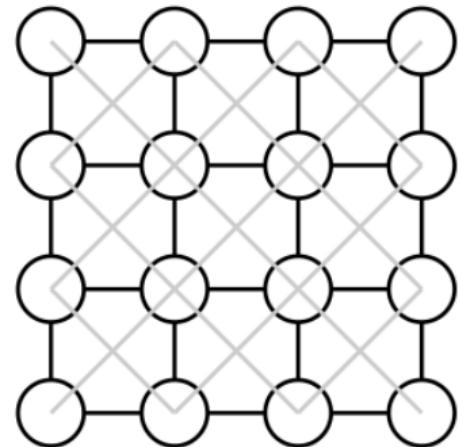
dwavesys.com

Embedding



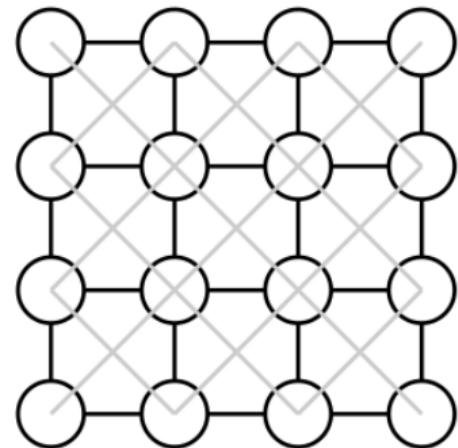
Embedding

- D-Wave's QPU kann als Graph U gesehen werden



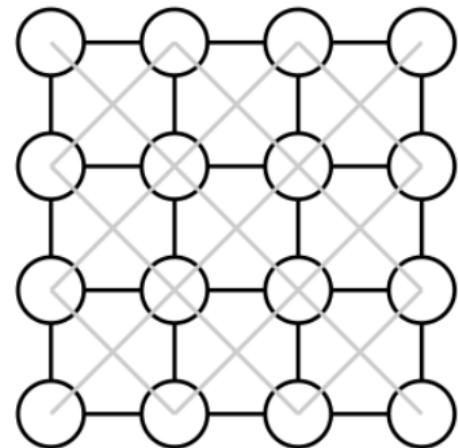
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$



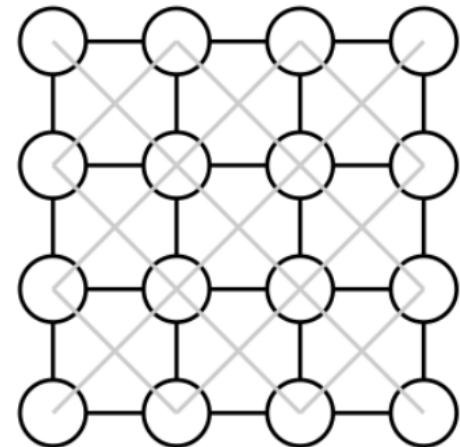
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$



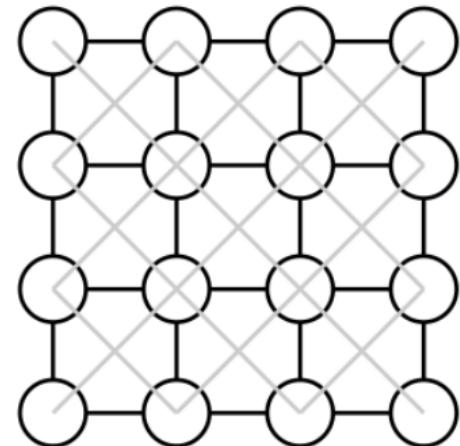
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$
- Gewichte:



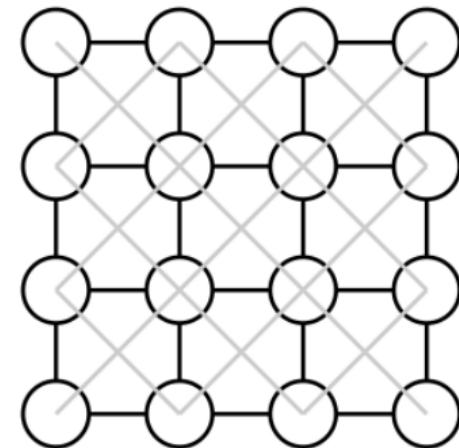
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$
- Gewichte:
 - $h_i(t)$ „qubit bias“
 - $\Delta_i(t)$ „tunneling amplitude“
 - $J_{ij}(t)$ „coupler strength“



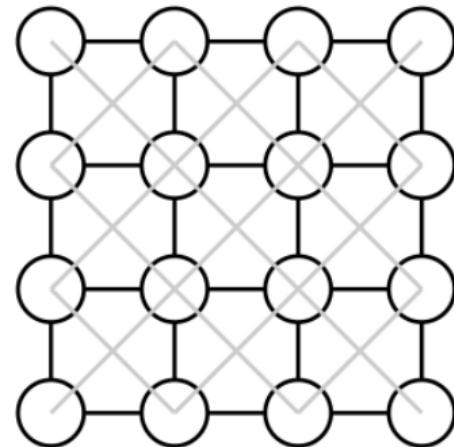
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$
- Gewichte:
 - $h_i(t)$ „qubit bias“
 - $\Delta_i(t)$ „tunneling amplitude“
 - $J_{ij}(t)$ „coupler strength“
- formuliere transversalen Ising-Hamiltonian auf Graph G
- $H(t) = \sum_{i \in V(G)} h_i(t) \sigma_i^z + \sum_{ij \in E(G)} J_{ij}(t) \sigma_i^z \sigma_j^z + \sum_{i \in V(G)} \Delta_i(t) \sigma_i^x$



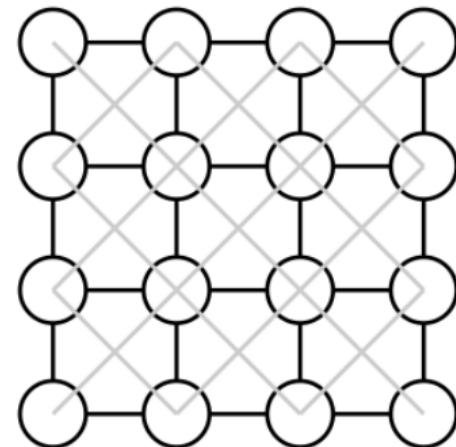
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$
- Gewichte:
 - $h_i(t)$ „qubit bias“
 - $\Delta_i(t)$ „tunneling amplitude“
 - $J_{ij}(t)$ „coupler strength“
- formuliere transversalen Ising-Hamiltonian auf Graph G
- $H(t) = \sum_{i \in V(G)} h_i(t) \sigma_i^z + \sum_{ij \in E(G)} J_{ij}(t) \sigma_i^z \sigma_j^z + \sum_{i \in V(G)} \Delta_i(t) \sigma_i^x$
- Eigenergie des Ising-Hamiltonians $\epsilon(s_1, \dots, s_n) = \sum_{i \in V(G)} h_i s_i + \sum_{ij \in E(G)} J_{ij} s_i s_j$



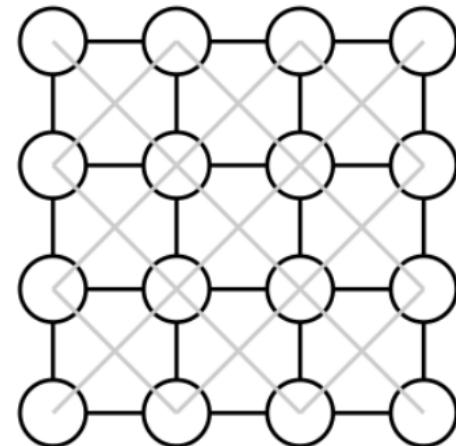
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$
- Gewichte:
 - $h_i(t)$ „qubit bias“
 - $\Delta_i(t)$ „tunneling amplitude“
 - $J_{ij}(t)$ „coupler strength“
- formuliere transversalen Ising-Hamiltonian auf Graph G
- $H(t) = \sum_{i \in V(G)} h_i(t) \sigma_i^z + \sum_{ij \in E(G)} J_{ij}(t) \sigma_i^z \sigma_j^z + \sum_{i \in V(G)} \Delta_i(t) \sigma_i^x$
- Eigenergie des Ising-Hamiltonians $\epsilon(s_1, \dots, s_n) = \sum_{i \in V(G)} h_i s_i + \sum_{ij \in E(G)} J_{ij} s_i s_j$
 - Spins $s_i \in \{-1, 1\}$



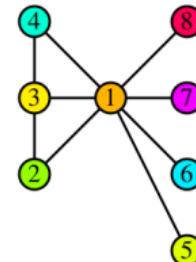
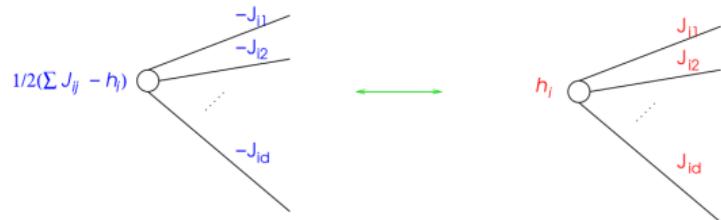
Embedding

- D-Wave's QPU kann als Graph U gesehen werden
 - Gewichtete Vertices (qubit) $i \in V(U)$
 - Gewichtete Kanten (coupler) $ij \in E(U)$
- Gewichte:
 - $h_i(t)$ „qubit bias“
 - $\Delta_i(t)$ „tunneling amplitude“
 - $J_{ij}(t)$ „coupler strength“
- formuliere transversalen Ising-Hamiltonian auf Graph G
- $H(t) = \sum_{i \in V(G)} h_i(t) \sigma_i^z + \sum_{ij \in E(G)} J_{ij}(t) \sigma_i^z \sigma_j^z + \sum_{i \in V(G)} \Delta_i(t) \sigma_i^x$
- Eigenergie des Ising-Hamiltonians $\epsilon(s_1, \dots, s_n) = \sum_{i \in V(G)} h_i s_i + \sum_{ij \in E(G)} J_{ij} s_i s_j$
 - Spins $s_i \in \{-1, 1\}$
- Suche nach Grundzustandsenergie ist das „Ising-Problem“



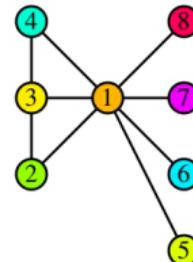
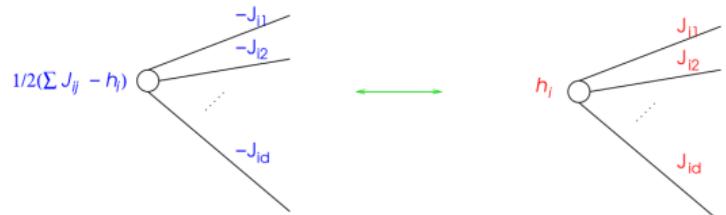
Embedding

- Ising- äquivalent zu QUBO-Problem auf dem selben Graph G



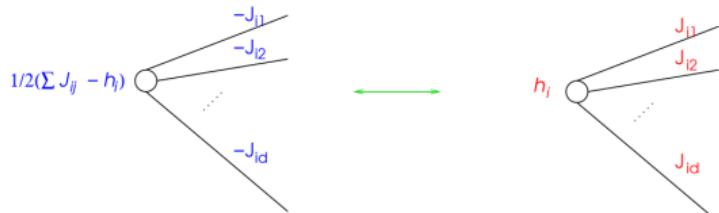
Embedding

- Ising- äquivalent zu QUBO-Problem auf dem selben Graph G
 - $Y(x_1, \dots, x_n) = \sum_{i \in V(G)} c_i x_i - \sum_{ij \in E(G)} J_{ij} x_i x_j$ maximieren
 - $x_i \in \{0, 1\}$



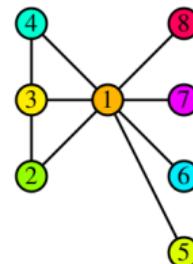
Embedding

- Ising- äquivalent zu QUBO-Problem auf dem selben Graph G
 - $Y(x_1, \dots, x_n) = \sum_{i \in V(G)} c_i x_i - \sum_{ij \in E(G)} J_{ij} x_i x_j$ maximieren
 - $x_i \in \{0, 1\}$



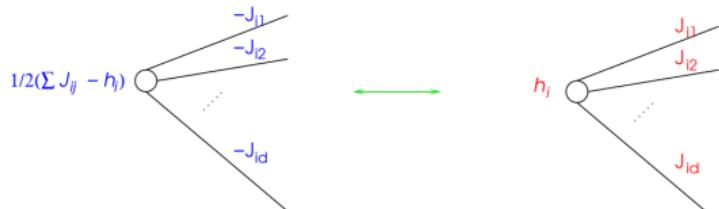
Ziel

G in U einbetten („subgraph-embedding“) und somit das Ising/QUBO-Problem lösen



Embedding

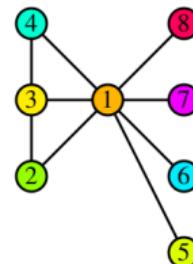
- Ising- äquivalent zu QUBO-Problem auf dem selben Graph G
 - $Y(x_1, \dots, x_n) = \sum_{i \in V(G)} c_i x_i - \sum_{ij \in E(G)} J_{ij} x_i x_j$ maximieren
 - $x_i \in \{0, 1\}$



Ziel

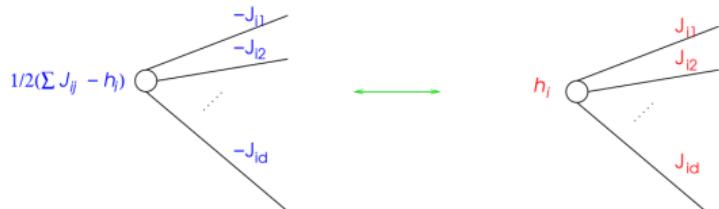
G in U einbetten („subgraph-embedding“) und somit das Ising/QUBO-Problem lösen

- Problem: Einschränkungen auf U (z.B. Anzahl der coupler) übertragen sich auf G



Embedding

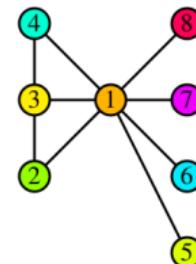
- Ising- äquivalent zu QUBO-Problem auf dem selben Graph G
 - $Y(x_1, \dots, x_n) = \sum_{i \in V(G)} c_i x_i - \sum_{ij \in E(G)} J_{ij} x_i x_j$ maximieren
 - $x_i \in \{0, 1\}$



Ziel

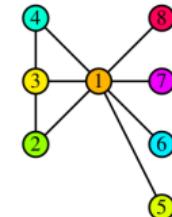
G in U einbetten („subgraph-embedding“) und somit das Ising/QUBO-Problem lösen

- Problem: Einschränkungen auf U (z.B. Anzahl der coupler) übertragen sich auf G
 - „dummy“-Vertices mit ferromagnetischer Kopplung einführen

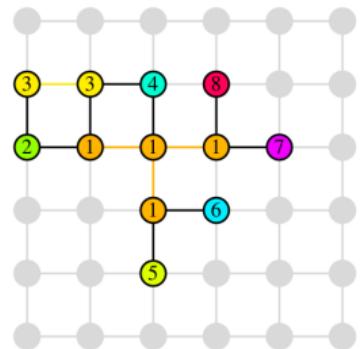


V. Choi, D-Wave Systems Inc. (2008)

Minor embedding



G

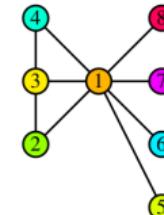


G_{emb}

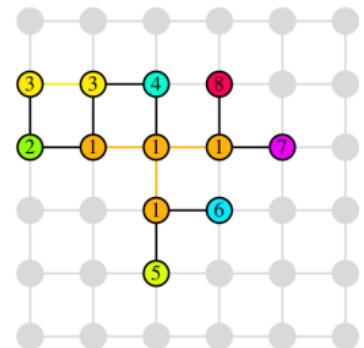
V. Choi, D-Wave Systems Inc. (2008)

Minor embedding

- logischer qubit i aus G wird auf subtree T_i physischer qubits aus $G_{\text{emb}} \subset U$ abgebildet



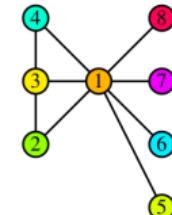
G



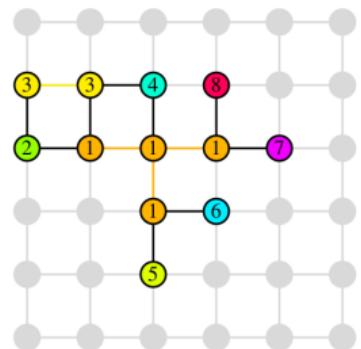
G_{emb}

Minor embedding

- logischer qubit i aus G wird auf subtree T_i physischer qubits aus $G_{\text{emb}} \subset U$ abgebildet
- ferromagnetische Kopplung zwischen „dummy“-Vertices muss hinreichend groß sein



G



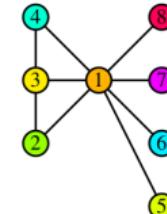
G_{emb}

Minor embedding

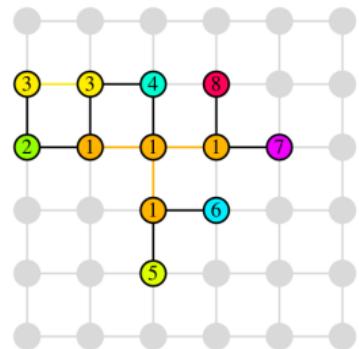
- logischer qubit i aus G wird auf subtree T_i physischer qubits aus $G_{\text{emb}} \subset U$ abgebildet
- ferromagnetische Kopplung zwischen „dummy“-Vertices muss hinreichend groß sein

Ziel

Parameter des minor-embedding auf ursprüngliche Ising/QUBO-Parameter zurückführen



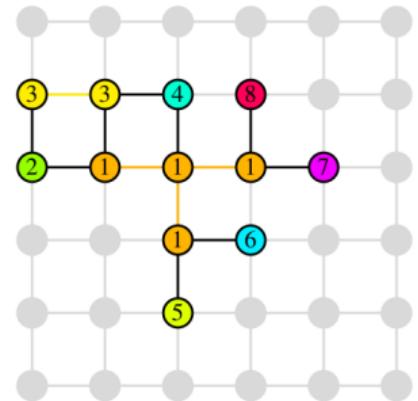
G



G_{emb}

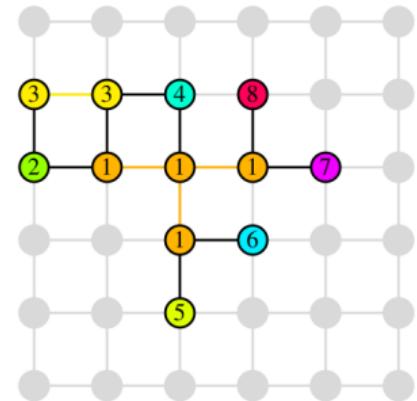
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren



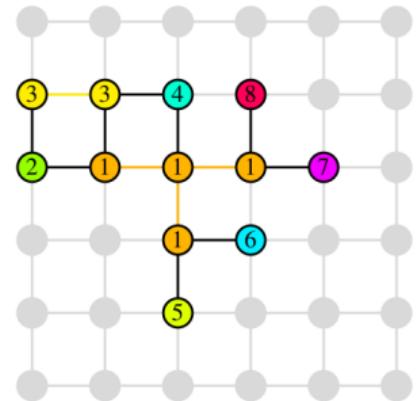
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
 - $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$



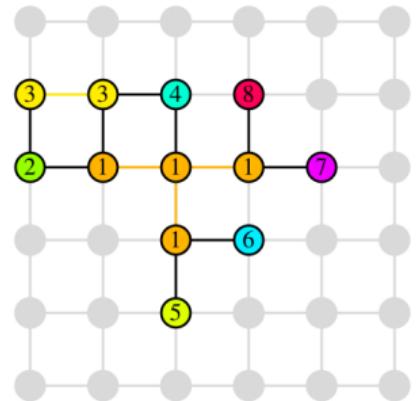
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
- $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$
- $\text{OE}(G_{\text{emb}}) = \cup_{ij \in E(G)} i_{\tau(i,j)} j_{\tau(j,i)}$ sind ursprüngliche coupler



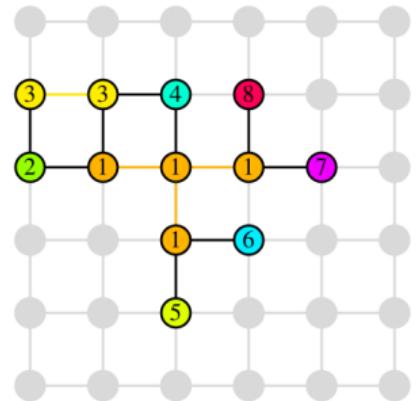
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
- $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$
- $\text{OE}(G_{\text{emb}}) = \cup_{ij \in E(G)} i_{\tau(i,j)} j_{\tau(j,i)}$ sind ursprüngliche coupler
 → $J'_{i_{\tau(j,i)} j_{\tau(j,i)}} = J_{ij}$ für $i_{\tau(i,j)} j_{\tau(j,i)} \in \text{OE}(G_{\text{emb}})$



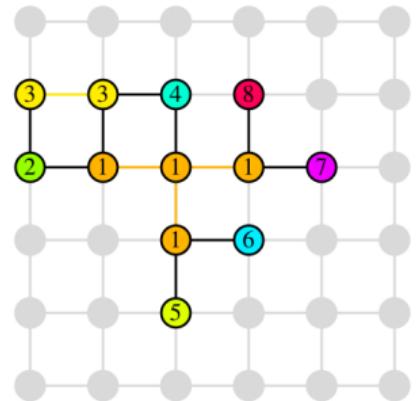
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
- $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$
- $\text{OE}(G_{\text{emb}}) = \cup_{ij \in E(G)} i_{\tau(i,j)} j_{\tau(j,i)}$ sind ursprüngliche coupler
→ $J'_{i_{\tau(i,j)} j_{\tau(j,i)}} = J_{ij}$ für $i_{\tau(i,j)} j_{\tau(j,i)} \in \text{OE}(G_{\text{emb}})$
- ferromagnetische coupler strength F_k^{ij} für $ij \in E(T_i)$



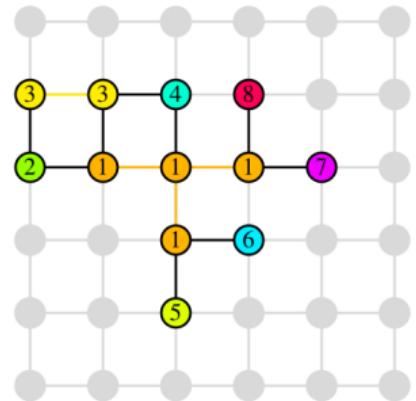
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
- $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$
- $\text{OE}(G_{\text{emb}}) = \cup_{ij \in E(G)} i_{\tau(i,j)} j_{\tau(j,i)}$ sind ursprüngliche coupler
 → $J'_{i_{\tau(j,i)} j_{\tau(j,i)}} = J_{ij}$ für $i_{\tau(i,j)} j_{\tau(j,i)} \in \text{OE}(G_{\text{emb}})$
- ferromagnetische coupler strength F_k^{ij} für $ij \in E(T_i)$
- bias h_i der logischen qubits auf physische qubits verteilen



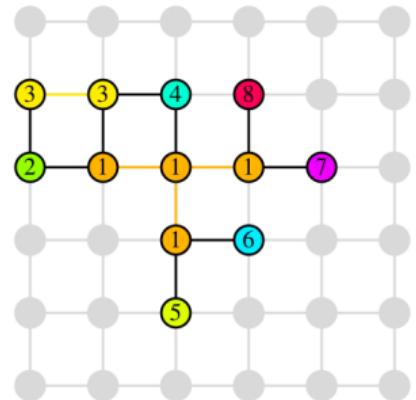
Minor embedding

- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
- $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$
- $\text{OE}(G_{\text{emb}}) = \cup_{ij \in E(G)} i_{\tau(i,j)} j_{\tau(j,i)}$ sind ursprüngliche coupler
 → $J'_{i_{\tau(j,i)} j_{\tau(j,i)}} = J_{ij}$ für $i_{\tau(i,j)} j_{\tau(j,i)} \in \text{OE}(G_{\text{emb}})$
- ferromagnetische coupler strength F_k^{ij} für $ij \in E(T_i)$
- bias h_i der logischen qubits auf physische qubits verteilen
 → $\sum_{i_k \in V(T_i)} h'_{i_k} = h_i$



Minor embedding

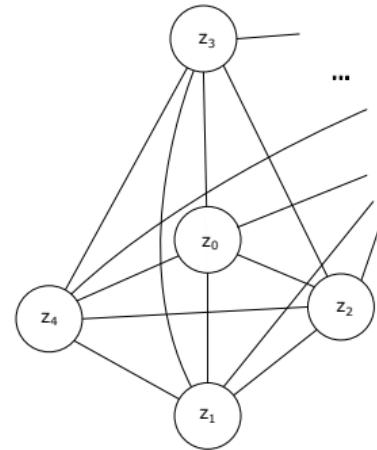
- $\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G_{\text{emb}})} h'_i s_i + \sum_{ij \in E(G_{\text{emb}})} J'_{ij} s_i s_j$ minimieren
- $\tau : V(G) \times V(G) \rightarrow V(U)$, so dass $i_{\tau(i,j)} \in V(T_i)$, $j_{\tau(j,i)} \in V(T_j)$ mit $i_{\tau(i,j)} j_{\tau(j,i)} \in E(U)$
- $\text{OE}(G_{\text{emb}}) = \cup_{ij \in E(G)} i_{\tau(i,j)} j_{\tau(j,i)}$ sind ursprüngliche coupler
 → $J'_{i_{\tau(j,i)} j_{\tau(j,i)}} = J_{ij}$ für $i_{\tau(i,j)} j_{\tau(j,i)} \in \text{OE}(G_{\text{emb}})$
- ferromagnetische coupler strength F_k^{ij} für $ij \in E(T_i)$
- bias h_i der logischen qubits auf physische qubits verteilen
 → $\sum_{i_k \in V(T_i)} h'_{i_k} = h_i$



Finale Energie

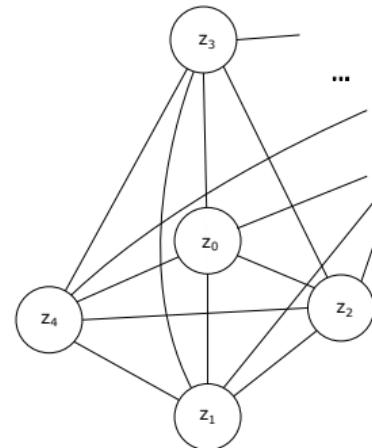
$$\epsilon^{\text{emb}}(s_1, \dots, s_N) = \sum_{i \in V(G)} \left(\sum_{i_k \in V(T_i)} h'_{i_k} s_{i_k} + \sum_{i_p i_q \in E(T_i)} F_i^{pq} s_{i_p} s_{i_q} \right) + \sum_{ij \in E(G)} J_{ij} s_{i_{\tau(i,j)}} s_{j_{\tau(i,j)}}$$

Fallbeispiel Aisin Corporation



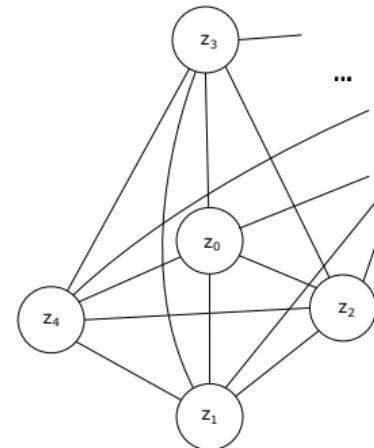
Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$



Fallbeispiel Aisin Corporation

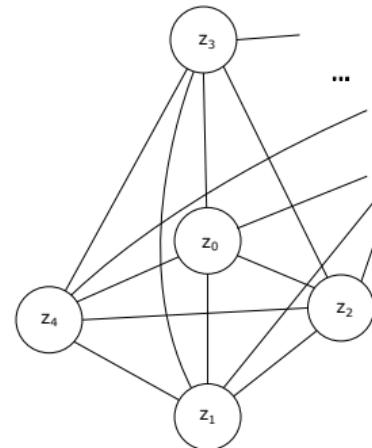
- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss



Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

Terme für QUBO-Formulierung:

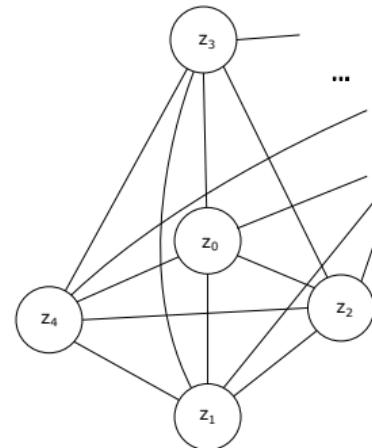


Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

Terme für QUBO-Formulierung:

- $f_{\text{local}}(\{x_i\}) = A_{\text{local}} \sum_t (1 - \sum_i x_{it})^2$

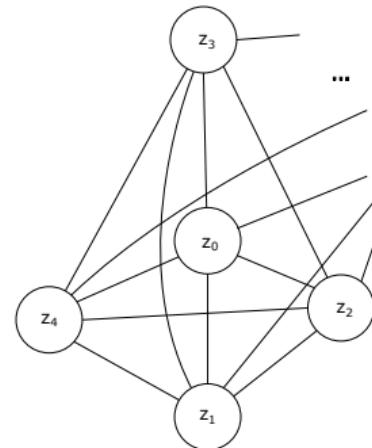


Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

Terme für QUBO-Formulierung:

- $f_{\text{local}}(\{x_i\}) = A_{\text{local}} \sum_t (1 - \sum_i x_{it})^2$
- $f_{\text{time}}(\{x_i\}) = A_{\text{time}} \sum_{ijt} T_{ij} x_{it} x_{jt+1}$

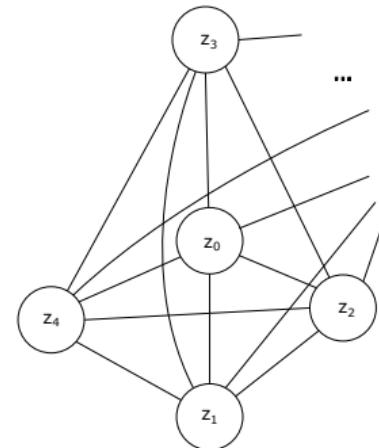


Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

Terme für QUBO-Formulierung:

- $f_{\text{local}}(\{x_i\}) = A_{\text{local}} \sum_t (1 - \sum_i x_{it})^2$
- $f_{\text{time}}(\{x_i\}) = A_{\text{time}} \sum_{ijt} T_{ij} x_{it} x_{jt+1}$
- $f_{\text{demand}}(\{x_i\}) = -A_{\text{demand}} \sum_{ijt} \sum_{\delta=1}^{\delta_*(t)} D_{ij} x_{it} x_{jt+\delta}$ mit $\delta_*(t) = \min(\delta_{\max}, \tau - 1 - t)$

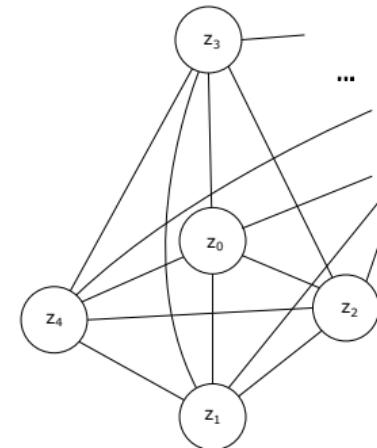


Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

Terme für QUBO-Formulierung:

- $f_{\text{local}}(\{x_i\}) = A_{\text{local}} \sum_t (1 - \sum_i x_{it})^2$
- $f_{\text{time}}(\{x_i\}) = A_{\text{time}} \sum_{ijt} T_{ij} x_{it} x_{jt+1}$
- $f_{\text{demand}}(\{x_i\}) = -A_{\text{demand}} \sum_{ijt} \sum_{\delta=1}^{\delta_*(t)} D_{ij} x_{it} x_{jt+\delta}$ mit $\delta_*(t) = \min(\delta_{\max}, \tau - 1 - t)$
- $f'_{\text{nonredundant}}(\{x_i\}) = A_{\text{nonredundant}} \sum_{ij} I_{ij} \sum_{\delta=2}^{\tau-2} \sum_{t=0}^{\tau-2-\delta} x_{it} x_{jt+1} x_{it+\delta} x_{jt+1+\delta}$

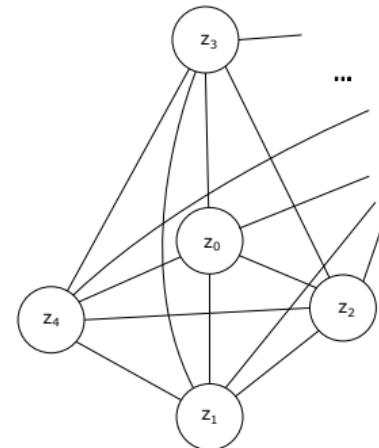


Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

Terme für QUBO-Formulierung:

- $f_{\text{local}}(\{x_i\}) = A_{\text{local}} \sum_t (1 - \sum_i x_{it})^2$
- $f_{\text{time}}(\{x_i\}) = A_{\text{time}} \sum_{ijt} T_{ij} x_{it} x_{jt+1}$
- $f_{\text{demand}}(\{x_i\}) = -A_{\text{demand}} \sum_{ijt} \sum_{\delta=1}^{\delta_*(t)} D_{ij} x_{it} x_{jt+\delta}$ mit $\delta_*(t) = \min(\delta_{\max}, \tau - 1 - t)$
- $f'_{\text{nonredundant}}(\{x_i\}) = A_{\text{nonredundant}} \sum_{ij} I_{ij} \sum_{\delta=2}^{\tau-2} \sum_{t=0}^{\tau-2-\delta} x_{it} x_{jt+1} x_{it+\delta} x_{jt+1+\delta}$
 - Vorsicht: Grad des Polynoms muss noch auf zwei reduziert werden



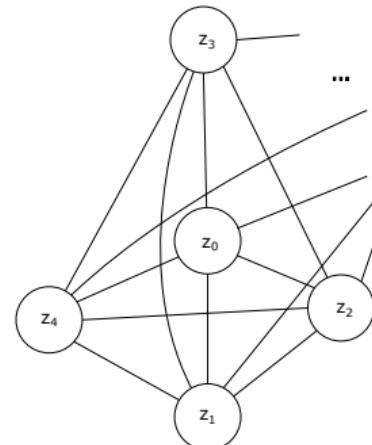
Fallbeispiel Aisin Corporation

- Zeit für Route ξ : $\text{time}(\xi) = \sum_{t=1}^{k-1} T_{\xi_t, \xi_{t+1}}$
- off-board-demand D_{ij} , der von i nach j gebracht werden muss

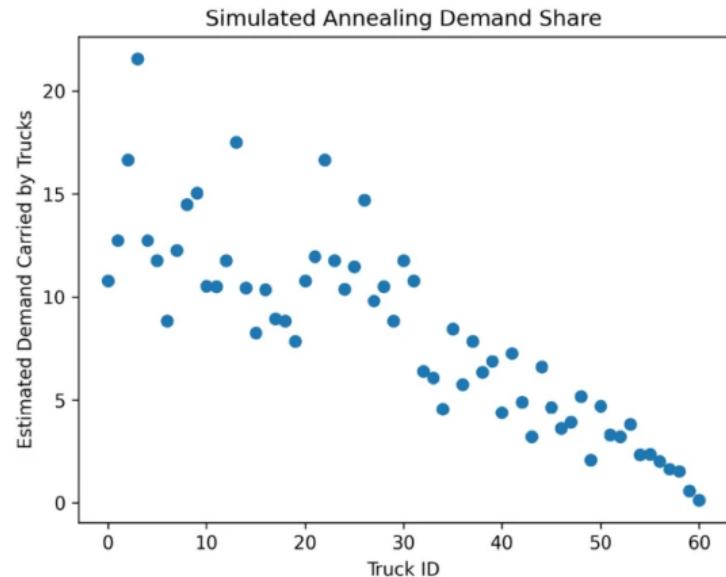
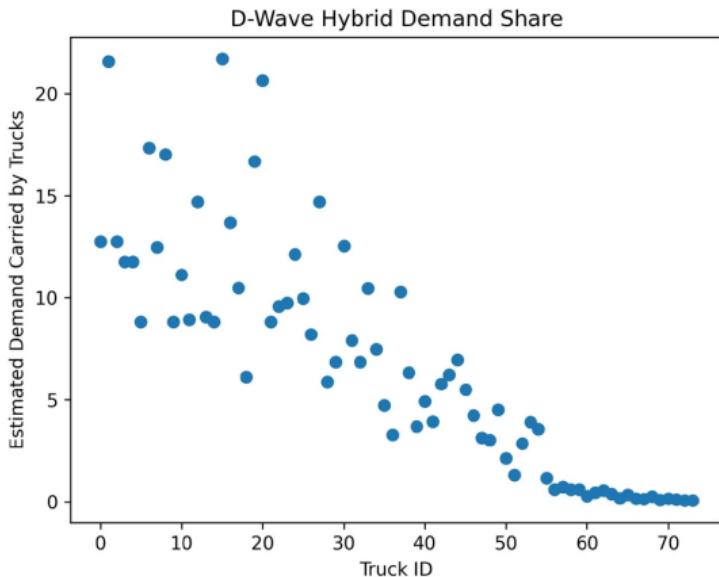
Terme für QUBO-Formulierung:

- $f_{\text{local}}(\{x_i\}) = A_{\text{local}} \sum_t (1 - \sum_i x_{it})^2$
- $f_{\text{time}}(\{x_i\}) = A_{\text{time}} \sum_{ijt} T_{ij} x_{it} x_{jt+1}$
- $f_{\text{demand}}(\{x_i\}) = -A_{\text{demand}} \sum_{ijt} \sum_{\delta=1}^{\delta_*(t)} D_{ij} x_{it} x_{jt+\delta}$ mit $\delta_*(t) = \min(\delta_{\max}, \tau - 1 - t)$
- $f'_{\text{nonredundant}}(\{x_i\}) = A_{\text{nonredundant}} \sum_{ij} I_{ij} \sum_{\delta=2}^{\tau-2} \sum_{t=0}^{\tau-2-\delta} x_{it} x_{jt+1} x_{it+\delta} x_{jt+1+\delta}$
 - Vorsicht: Grad des Polynoms muss noch auf zwei reduziert werden

$$A_{\text{local}} = 5000, \quad A_{\text{demand}} = 320, \quad A_{\text{time}} = 0.01, \quad A_{\text{nonredundant}} = 1$$

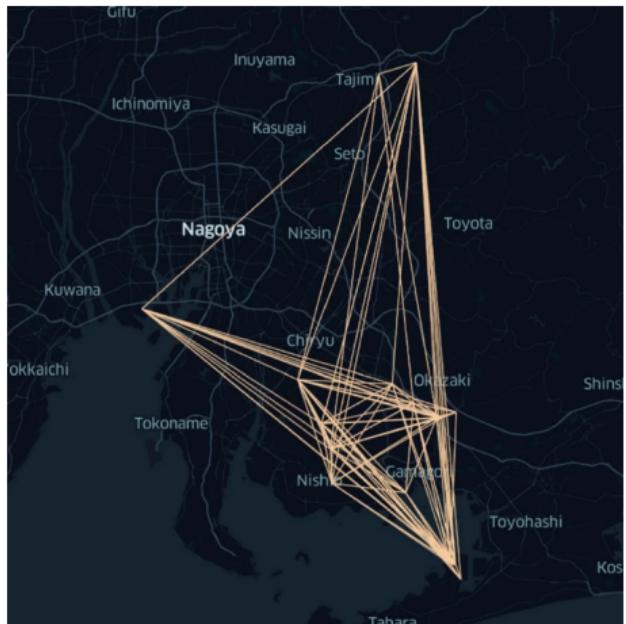
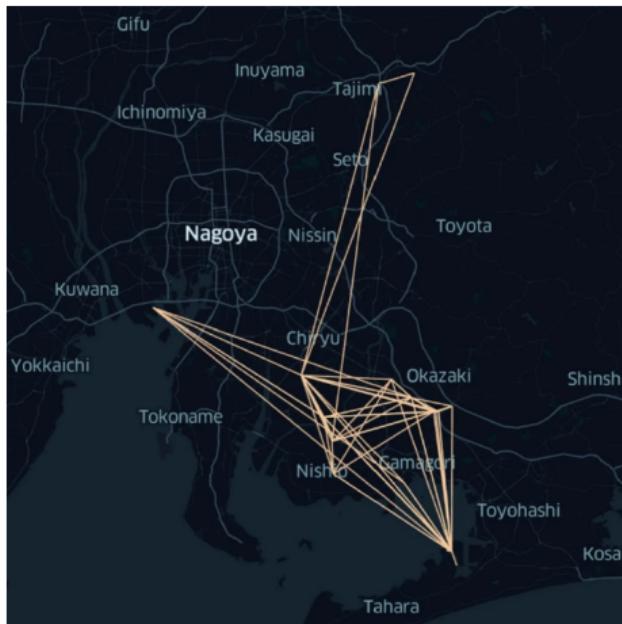


Fallbeispiel Aisin Corporation



Weinberg, S.J., Sanches, F., Ide, T. et al. Sci Rep 13, 4770 (2023)

Fallbeispiel Aisin Corporation



Weinberg, S.J., Sanches, F., Ide, T. et al. Sci Rep 13, 4770 (2023)

Zusammenfassung und Ausblick

- QA bereits angewandt um Logistik (deutlich) zu verbessern
- QA konvergiert wahrscheinlicher als SA
- nur auf kombinatorische Optimierungsprobleme anwendbar
- nur eine heuristische Methode
- Komplexität wächst bei QA exponentiell mit Systemgröße

- mehr qubits
 - größere Probleme optimieren
- mehr coupler
 - weniger „dummy“-Vertices nötig
 - weniger physische qubits für einen logischen qubit nötig

Quellen

- Bapst, V. et al., The quantum adiabatic algorithm applied to random optimization problems: The quantum Spin glass perspective, ScienceDirect 523 (2013), <https://doi.org/10.1016/j.physrep.2012.10.002>
- Kadowaki, T. et al. Quantum annealing in the transverse Ising model, PHYSICAL REVIEW E 58 (1998), <https://doi.org/10.48550/arXiv.cond-mat/9804280>
- Rieffel, E.G., Venturelli, D., O’Gorman, B. et al. A case study in programming a quantum annealer for hard operational planning problems. Quantum Inf Process 14, 1–36 (2015). <https://doi.org/10.1007/s11128-014-0892-x>
- Choi, V. Minor-embedding in adiabatic quantum computation: I. The parameter setting problem. Quantum Inf Process 7, 193–209 (2008). <https://doi.org/10.1007/s11128-008-0082-9>
- Jain, S, Solving the Traveling Salesman Problem on the D-Wave Quantum Computer, Frontiers in Physics, Volume 9, id.646 (2021) ,doi: 10.3389/fphy.2021.760783
- Sangwan, Shabnam. (2018). Literature Review on Travelling Salesman Problem. International Journal of Research. 5. 1152.
- Weinberg, S.J., Sanches, F., Ide, T. et al. Supply chain logistics with quantum and classical annealing algorithms. Sci Rep 13, 4770 (2023). <https://doi.org/10.1038/s41598-023-31765-8>
- Yarkoni, S et al., Quantum annealing for industry applications: introduction and review, IOP Publishing 85-10 (2022), <https://dx.doi.org/10.1088/1361-6633/ac8c54>

Backup-Folien für Detailfragen

Details, evtl. Rechnungen und Zahlenwerte für Fragen