Quantum Annealing for Music Arrangement

Lucas Kirby

4 March 2025

Department of Physics, Durham University

2025-03-09

Quantum Annealing for Music Arrangement

Quantum Annealing
for Music Arrangement

Lucas Kirby
4 March 2025
Department of Physics, Durham University

Overview

Theory

Adiabatic quantum computing

Quantum annealing

Motivations

Music arrangement

Method

Results

Conclusions

Overview

Theory
Adiabatic quantum computing
Quantum annealing
Motivations
Music arrangement
Method
Results
Conclusions

- AQC, more general umbrella term for the technique
- Quantum annealing as a subset of AQC and what that involves
- Music arrangement and why we're looking at this problem
- How the problem is solved, and the following results
- Conclusions and future work

Theory

Theory

Adiabatic quantum computing

Adiabatic principle

A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]

[1] Born and Fock, 'Beweis des Adiabatensatzes'.

Adiabatic quantum computing
Adiabatic principle
A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]
[1] Born and Fock, 'Beweis des Adiabatensatzes'.

- Adiabatic principle — system remains in the same eigenstate if perturbed slowly enough (without transferring heat)
- Equation shows evolution from initial Hamiltonian H_0 to final H_p over time T
- Importantly, if the system starts in the ground state, it will end in the ground state
- Impossible in practice as true adiabatic evolution would take infinite time, infinitely many steps

Adiabatic quantum computing

Adiabatic principle

A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]

$$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$$

[1] Born and Fock, 'Beweis des Adiabatensatzes'.

Adiabatic quantum computing
Adiabatic principle
A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]
$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$
[1] Born and Fock, 'Beweis des Adiabatensatzes'.

- Adiabatic principle — system remains in the same eigenstate if perturbed slowly enough (without transferring heat)
- Equation shows evolution from initial Hamiltonian H_0 to final H_p over time T
- Importantly, if the system starts in the ground state, it will end in the ground state
- Impossible in practice as true adiabatic evolution would take infinite time, infinitely many steps

Adiabatic quantum computing

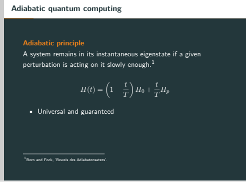
Adiabatic principle

A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]

$$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$$

- Universal and guaranteed

[1] Born and Fock, ‘Beweis des Adiabatensatzes’.



- Adiabatic principle — system remains in the same eigenstate if perturbed slowly enough (without transferring heat)
- Equation shows evolution from initial Hamiltonian H_0 to final H_p over time T
- Importantly, if the system starts in the ground state, it will end in the ground state
- Impossible in practice as true adiabatic evolution would take infinite time, infinitely many steps

Adiabatic quantum computing

Adiabatic principle

A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]

$$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$$

- Universal and guaranteed
- A system that starts in a ground state, ends in a ground state

[1] Born and Fock, 'Beweis des Adiabatensatzes'.

Adiabatic quantum computing
Adiabatic principle
A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]
$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$
- Universal and guaranteed
- A system that starts in a ground state, ends in a ground state
[1] Born and Fock, 'Beweis des Adiabatensatzes'.

- Adiabatic principle — system remains in the same eigenstate if perturbed slowly enough (without transferring heat)
- Equation shows evolution from initial Hamiltonian H_0 to final H_p over time T
- Importantly, if the system starts in the ground state, it will end in the ground state
- Impossible in practice as true adiabatic evolution would take infinite time, infinitely many steps

Adiabatic quantum computing

Adiabatic principle

A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]

$$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$$

- Universal and guaranteed
- A system that starts in a ground state, ends in a ground state
- Not possible in practice

[1] Born and Fock, 'Beweis des Adiabatensatzes'.

Adiabatic quantum computing

Adiabatic principle

A system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough.[1]

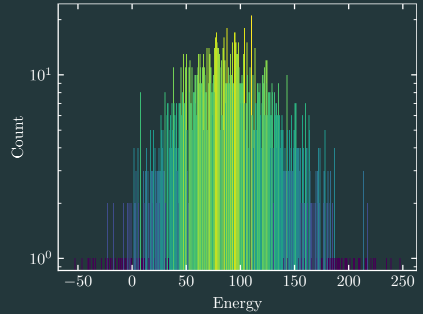
$$H(t) = \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_p$$

- Universal and guaranteed
- A system that starts in a ground state, ends in a ground state
- Not possible in practice

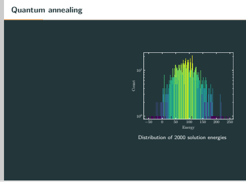
[1] Born and Fock, 'Beweis des Adiabatensatzes'.

- Adiabatic principle — system remains in the same eigenstate if perturbed slowly enough (without transferring heat)
- Equation shows evolution from initial Hamiltonian H_0 to final H_p over time T
- Importantly, if the system starts in the ground state, it will end in the ground state
- Impossible in practice as true adiabatic evolution would take infinite time, infinitely many steps

Quantum annealing



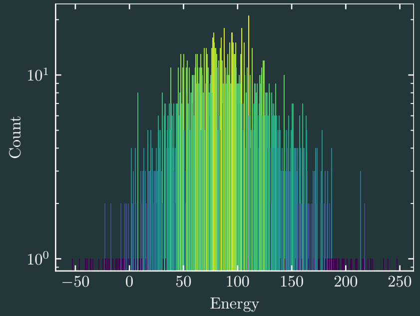
Distribution of 2000 solution energies



- Subset of AQC, relaxes the adiabaticity condition
- Annealing — slow heating of a material to change its properties
- Evolution time shortened (order of a few μs)
- End state no longer guaranteed, if started in ground state could end in excited state
- Able to run the evolution many times
- Probabilistic distribution of outcomes, sometimes will get lucky

Quantum annealing

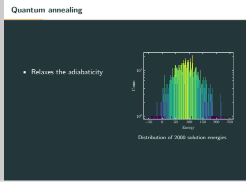
- Relaxes the adiabaticity



Distribution of 2000 solution energies

2025-03-09

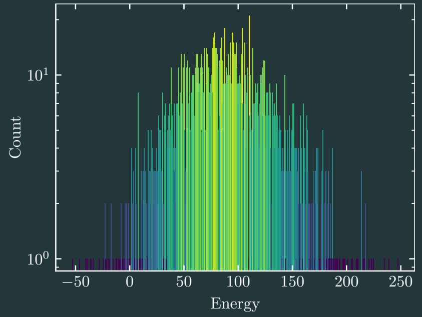
Quantum Annealing for Music Arrangement
└Theory
  └Quantum annealing
    └Quantum annealing



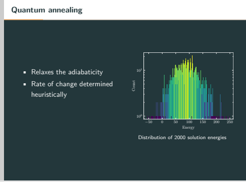
- Subset of AQC, relaxes the adiabaticity condition
- Annealing — slow heating of a material to change its properties
- Evolution time shortened (order of a few μs)
- End state no longer guaranteed, if started in ground state could end in excited state
- Able to run the evolution many times
- Probabilistic distribution of outcomes, sometimes will get lucky

Quantum annealing

- Relaxes the adiabaticity
- Rate of change determined heuristically



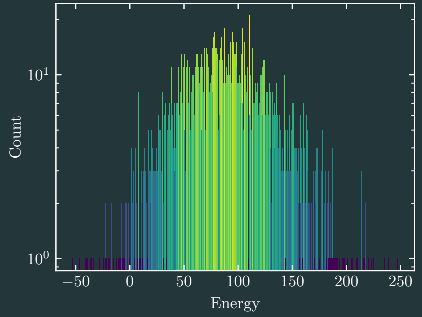
Distribution of 2000 solution energies



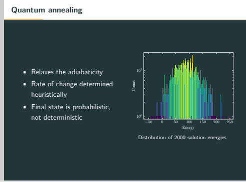
- Subset of AQC, relaxes the adiabaticity condition
- Annealing — slow heating of a material to change its properties
- Evolution time shortened (order of a few μs)
- End state no longer guaranteed, if started in ground state could end in excited state
- Able to run the evolution many times
- Probabilistic distribution of outcomes, sometimes will get lucky

Quantum annealing

- Relaxes the adiabaticity
- Rate of change determined heuristically
- Final state is probabilistic, not deterministic

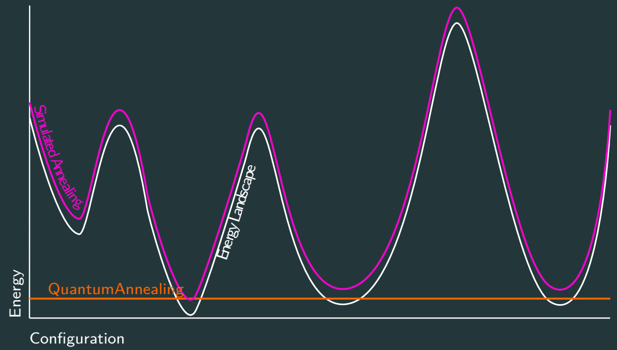


Distribution of 2000 solution energies

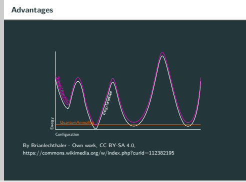


- Subset of AQC, relaxes the adiabaticity condition
- Annealing — slow heating of a material to change its properties
- Evolution time shortened (order of a few μs)
- End state no longer guaranteed, if started in ground state could end in excited state
- Able to run the evolution many times
- Probabilistic distribution of outcomes, sometimes will get lucky

Advantages



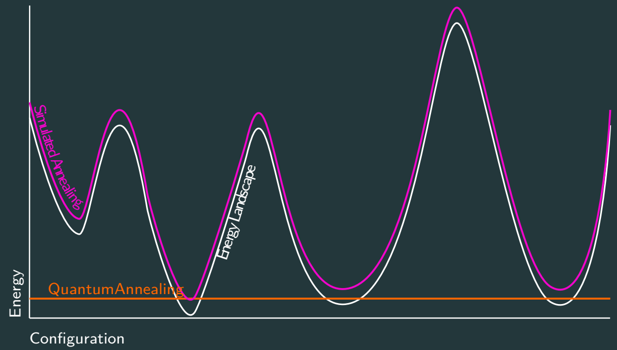
By Brianlechthaler - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=112382195

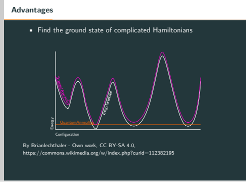


- Why is this technique useful?
- Allows us to find the ground state of complicated Hamiltonians by starting from an easy one
- Diagram — energy against configuration space, simulated annealing (classical) traverses the "energy landscape" whereas quantum annealing tunnels through it
- As opposed to classical methods, does not get affected by local minima
- Technique very good for solving optimisation problems e.g. travelling salesman, with complicated energy landscapes

Advantages

- Find the ground state of complicated Hamiltonians

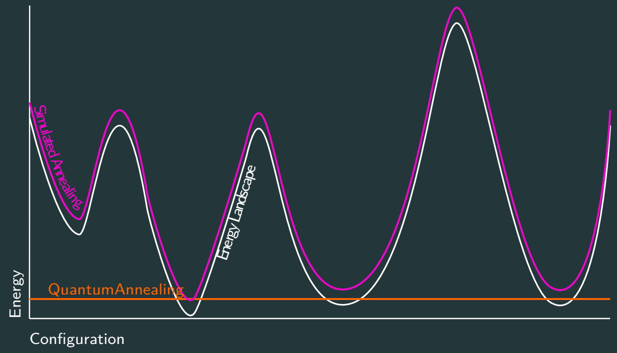


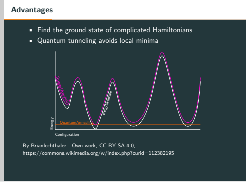


- Why is this technique useful?
- Allows us to find the ground state of complicated Hamiltonians by starting from an easy one
- Diagram — energy against configuration space, simulated annealing (classical) traverses the "energy landscape" whereas quantum annealing tunnels through it
- As opposed to classical methods, does not get affected by local minima
- Technique very good for solving optimisation problems e.g. travelling salesman, with complicated energy landscapes

Advantages

- Find the ground state of complicated Hamiltonians
- Quantum tunneling avoids local minima





- Why is this technique useful?
- Allows us to find the ground state of complicated Hamiltonians by starting from an easy one
- Diagram — energy against configuration space, simulated annealing (classical) traverses the "energy landscape" whereas quantum annealing tunnels through it
- As opposed to classical methods, does not get affected by local minima
- Technique very good for solving optimisation problems e.g. travelling salesman, with complicated energy landscapes

Ising model

Lattice of variables with two discrete values

Ising model

Lattice of variables with two discrete values

- How can we model the Hamiltonians?
- Ising model, a lattice of variables with two discrete values (+1/-1), acted on by spin operators σ
- Start with initial Hamiltonian, superposition of all possible states, easy to prepare and find the ground state
- Problem Hamiltonian, coupling strengths J_{ij} and field strengths h_i, describe interactions (biases) of the spins
- Want to encode the problem solution into the ground state of this Hamiltonian so that the system will give the solution after evolution

Ising model

Lattice of variables with two discrete values

Initial Hamiltonian

$$H_0 = h_0 \sum_{i=1}^{N} \sigma_i^x$$

Ising model

Lattice of variables with two discrete values

Initial Hamiltonian

$$H_0 = h_0 \sum_{i=1}^{N} \sigma_i^x$$

- How can we model the Hamiltonians?
- Ising model, a lattice of variables with two discrete values (+1/-1), acted on by spin operators σ
- Start with initial Hamiltonian, superposition of all possible states, easy to prepare and find the ground state
- Problem Hamiltonian, coupling strengths J_{ij} and field strengths h_i, describe interactions (biases) of the spins
- Want to encode the problem solution into the ground state of this Hamiltonian so that the system will give the solution after evolution

Ising model

Lattice of variables with two discrete values

Initial Hamiltonian

$$H_0 = h_0 \sum_{i=1}^{N} \sigma_i^x$$

Problem Hamiltonian

$$H_p = \sum_{i<j}^{N} J_{ij} \sigma_i^z \sigma_j^z + \sum_{i=1}^{N} h_i \sigma_i^z$$

Ising model
Lattice of variables with two discrete values
Initial Hamiltonian
$H_0 = h_0 \sum_{i=1}^{N} \sigma_i^x$
Problem Hamiltonian
$H_p = \sum_{i<j}^{N} J_{ij} \sigma_i^z \sigma_j^z + \sum_{i=1}^{N} h_i \sigma_i^z$

- How can we model the Hamiltonians?
- Ising model, a lattice of variables with two discrete values (+1/-1), acted on by spin operators σ
- Start with initial Hamiltonian, superposition of all possible states, easy to prepare and find the ground state
- Problem Hamiltonian, coupling strengths J_{ij} and field strengths h_i, describe interactions (biases) of the spins
- Want to encode the problem solution into the ground state of this Hamiltonian so that the system will give the solution after evolution

QUBO

QUBO

- How to encode a problem into a Hamiltonian?
- Similar form to the Ising model, but with binary variables (0/1)
- Minimisation of this function should be the problem solution
- Set of binary variables x, matrix Q of real weights that describes interactions between variables
- After evolution, can read out the values of x to give solution

QUBO

Quadratic Unconstrained Binary Optimisation

Vector x of qubits, matrix Q of weights

$$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

QUBO

Quadratic Unconstrained Binary Optimisation
Vector x of qubits, matrix Q of weights
$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$

- How to encode a problem into a Hamiltonian?
- Similar form to the Ising model, but with binary variables (0/1)
- Minimisation of this function should be the problem solution
- Set of binary variables x, matrix Q of real weights that describes interactions between variables
- After evolution, can read out the values of x to give solution

QUBO

Quadratic Unconstrained Binary Optimisation

Vector x of qubits, matrix Q of weights

$$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

QUBO
Quadratic Unconstrained Binary Optimisation
Vector x of qubits, matrix Q of weights
$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$

- How to encode a problem into a Hamiltonian?
- Similar form to the Ising model, but with binary variables (0/1)
- Minimisation of this function should be the problem solution
- Set of binary variables x, matrix Q of real weights that describes interactions between variables
- After evolution, can read out the values of x to give solution

QUBO

Quadratic Unconstrained Binary Optimisation

Vector x of qubits, matrix Q of weights

$$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

QUBO
Quadratic Unconstrained Binary Optimisation
Vector x of qubits, matrix Q of weights
$$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

- How to encode a problem into a Hamiltonian?
- Similar form to the Ising model, but with binary variables (0/1)
- Minimisation of this function should be the problem solution
- Set of binary variables x, matrix Q of real weights that describes interactions between variables
- After evolution, can read out the values of x to give solution

QUBO

Quadratic Unconstrained Binary Optimisation

Vector x of qubits, matrix Q of weights

$$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

- Aim to minimise this function
- Difficult to solve analytically
- Mapped to H_p using simple change of variable
- Encodes problem solution into Hamiltonian's ground state

QUBO

Quadratic Unconstrained Binary Optimisation

Vector x of qubits, matrix Q of weights

$$f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

- Aim to minimise this function
- Difficult to solve analytically
- Mapped to H_p using simple change of variable
- Encodes problem solution into Hamiltonian's ground state

- How to encode a problem into a Hamiltonian?
- Similar form to the Ising model, but with binary variables (0/1)
- Minimisation of this function should be the problem solution
- Set of binary variables x, matrix Q of real weights that describes interactions between variables
- After evolution, can read out the values of x to give solution

Motivations

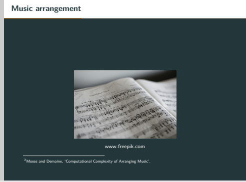
What problems can we solve?

What problems can we solve?

Music arrangement

www.freepik.com

[2] Moses and Demaine, 'Computational Complexity of Arranging Music'.



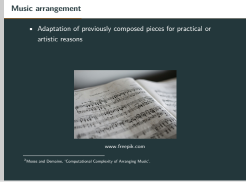
- Adaptation of music in terms of instrumentation, medium, or style
- Traditionally a complex process that requires a deep understanding of musical theory and structure
- Reduction is the rewriting of music for a smaller number of instruments (for example string quartet)
- Very large configuration space, many different combinations of notes that could produce the final arrangement
- For those interested, NP-hard in computational complexity theory, cannot be solved in polynomial time
- NB: all scores shown are own reproductions from public domain files

Music arrangement

- Adaptation of previously composed pieces for practical or artistic reasons

www.freepik.com

[2]Moses and Demaine, ‘Computational Complexity of Arranging Music’.



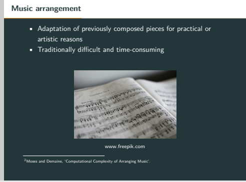
- Adaptation of music in terms of instrumentation, medium, or style
- Traditionally a complex process that requires a deep understanding of musical theory and structure
- Reduction is the rewriting of music for a smaller number of instruments (for example string quartet)
- Very large configuration space, many different combinations of notes that could produce the final arrangement
- For those interested, NP-hard in computational complexity theory, cannot be solved in polynomial time
- NB: all scores shown are own reproductions from public domain files

Music arrangement

- Adaptation of previously composed pieces for practical or artistic reasons
- Traditionally difficult and time-consuming

www.freepik.com

[2]Moses and Demaine, ‘Computational Complexity of Arranging Music’.



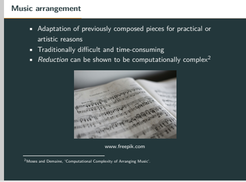
- Adaptation of music in terms of instrumentation, medium, or style
- Traditionally a complex process that requires a deep understanding of musical theory and structure
- Reduction is the rewriting of music for a smaller number of instruments (for example string quartet)
- Very large configuration space, many different combinations of notes that could produce the final arrangement
- For those interested, NP-hard in computational complexity theory, cannot be solved in polynomial time
- NB: all scores shown are own reproductions from public domain files

Music arrangement

- Adaptation of previously composed pieces for practical or artistic reasons
- Traditionally difficult and time-consuming
- *Reduction* can be shown to be computationally complex[2]

www.freepik.com

[2] Moses and Demaine, 'Computational Complexity of Arranging Music'.



- Adaptation of music in terms of instrumentation, medium, or style
- Traditionally a complex process that requires a deep understanding of musical theory and structure
- Reduction is the rewriting of music for a smaller number of instruments (for example string quartet)
- Very large configuration space, many different combinations of notes that could produce the final arrangement
- For those interested, NP-hard in computational complexity theory, cannot be solved in polynomial time
- NB: all scores shown are own reproductions from public domain files

Motivations

[3] Huang, Chiu and Shan, ‘Towards an automatic music arrangement framework using score reduction’; Nakamura and Yoshii, ‘Statistical piano reduction controlling performance difficulty’; Li et al., ‘Automatic Piano Reduction of Orchestral Music Based on Musical Entropy’.

[4] Freedline, ‘Algorhythms’; Arya et al., ‘Music Composition Using Quantum Annealing’.

[5] Miranda, *Quantum Computer Music*.

- Context of previous work
- Classical methods — machine learning, statistical analysis, rule-based systems, iterative and slow
- Applying quantum computing to music in the last five years, still a very young technology with limitations
- Has been used to generate music, not arrange it
- Methods shown here have not been found in the literature

Motivations

- Already exist classical methods of automatic arrangement[3]

[3] Huang, Chiu and Shan, ‘Towards an automatic music arrangement framework using score reduction’; Nakamura and Yoshii, ‘Statistical piano reduction controlling performance difficulty’; Li et al., ‘Automatic Piano Reduction of Orchestral Music Based on Musical Entropy’.

[4] Freedline, ‘Algorhythms’; Arya et al., ‘Music Composition Using Quantum Annealing’.

[5] Miranda, *Quantum Computer Music*.

Motivations
Already exist classical methods of automatic arrangement[3]

- Context of previous work
- Classical methods — machine learning, statistical analysis, rule-based systems, iterative and slow
- Applying quantum computing to music in the last five years, still a very young technology with limitations
- Has been used to generate music, not arrange it
- Methods shown here have not been found in the literature

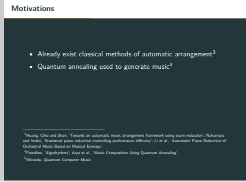
Motivations

- Already exist classical methods of automatic arrangement[3]
- Quantum annealing used to generate music[4]

[3] Huang, Chiu and Shan, 'Towards an automatic music arrangement framework using score reduction'; Nakamura and Yoshii, 'Statistical piano reduction controlling performance difficulty'; Li et al., 'Automatic Piano Reduction of Orchestral Music Based on Musical Entropy'.

[4] Freedline, 'Algorhythms'; Arya et al., 'Music Composition Using Quantum Annealing'.

[5] Miranda, *Quantum Computer Music*.



- Context of previous work
- Classical methods — machine learning, statistical analysis, rule-based systems, iterative and slow
- Applying quantum computing to music in the last five years, still a very young technology with limitations
- Has been used to generate music, not arrange it
- Methods shown here have not been found in the literature

Motivations

- Already exist classical methods of automatic arrangement[3]
- Quantum annealing used to generate music[4]
- Field of quantum computer music is very new[5]

[3] Huang, Chiu and Shan, 'Towards an automatic music arrangement framework using score reduction'; Nakamura and Yoshii, 'Statistical piano reduction controlling performance difficulty'; Li et al., 'Automatic Piano Reduction of Orchestral Music Based on Musical Entropy'.

[4] Freedline, 'Algorhythms'; Arya et al., 'Music Composition Using Quantum Annealing'.

[5] Miranda, *Quantum Computer Music*.

- Context of previous work
- Classical methods — machine learning, statistical analysis, rule-based systems, iterative and slow
- Applying quantum computing to music in the last five years, still a very young technology with limitations
- Has been used to generate music, not arrange it
- Methods shown here have not been found in the literature

Motivations

- Already exist classical methods of automatic arrangement[3]
- Quantum annealing used to generate music[4]
- Field of quantum computer music is very new[5]
- Novel adaption of this method to a new problem

[3] Huang, Chiu and Shan, 'Towards an automatic music arrangement framework using score reduction'; Nakamura and Yoshii, 'Statistical piano reduction controlling performance difficulty'; Li et al., 'Automatic Piano Reduction of Orchestral Music Based on Musical Entropy'.

[4] Freedline, 'Algorhythms'; Arya et al., 'Music Composition Using Quantum Annealing'.

[5] Miranda, *Quantum Computer Music*.

Motivations

- Already exist classical methods of automatic arrangement[3]
- Quantum annealing used to generate music[4]
- Field of quantum computer music is very new[5]
- Novel adaption of this method to a new problem

- Context of previous work
- Classical methods — machine learning, statistical analysis, rule-based systems, iterative and slow
- Applying quantum computing to music in the last five years, still a very young technology with limitations
- Has been used to generate music, not arrange it
- Methods shown here have not been found in the literature

Motivations

- Already exist classical methods of automatic arrangement[3]
- Quantum annealing used to generate music[4]
- Field of quantum computer music is very new[5]
- Novel adaption of this method to a new problem
- *This has never been done before!*

[3] Huang, Chiu and Shan, 'Towards an automatic music arrangement framework using score reduction'; Nakamura and Yoshii, 'Statistical piano reduction controlling performance difficulty'; Li et al., 'Automatic Piano Reduction of Orchestral Music Based on Musical Entropy'.

[4] Freedline, 'Algorhythms'; Arya et al., 'Music Composition Using Quantum Annealing'.

[5] Miranda, *Quantum Computer Music*.

Motivations

- Already exist classical methods of automatic arrangement[3]
- Quantum annealing used to generate music[4]
- Field of quantum computer music is very new[5]
- Novel adaption of this method to a new problem
- *This has never been done before!*

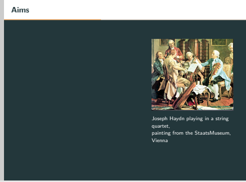
- Context of previous work
- Classical methods — machine learning, statistical analysis, rule-based systems, iterative and slow
- Applying quantum computing to music in the last five years, still a very young technology with limitations
- Has been used to generate music, not arrange it
- Methods shown here have not been found in the literature

Method

Method

Aims

Joseph Haydn playing in a string quartet,
painting from the StaatsMuseum, Vienna

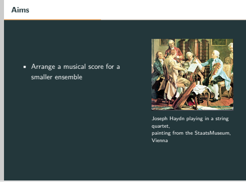


- What are we trying to do? What are the constraints to our problem?
- Take a musical score and reduce it to a smaller ensemble
- All notes must be taken from the original score, no new notes can be added
- Each instrument can only take notes from one part at a time

Aims

- Arrange a musical score for a smaller ensemble

Joseph Haydn playing in a string quartet,
painting from the StaatsMuseum, Vienna

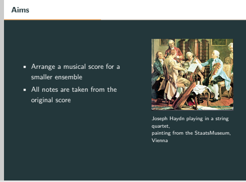


- What are we trying to do? What are the constraints to our problem?
- Take a musical score and reduce it to a smaller ensemble
- All notes must be taken from the original score, no new notes can be added
- Each instrument can only take notes from one part at a time

Aims

- Arrange a musical score for a smaller ensemble
- All notes are taken from the original score

Joseph Haydn playing in a string quartet,
painting from the StaatsMuseum, Vienna

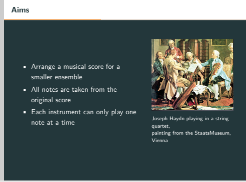


- What are we trying to do? What are the constraints to our problem?
- Take a musical score and reduce it to a smaller ensemble
- All notes must be taken from the original score, no new notes can be added
- Each instrument can only take notes from one part at a time

Aims

- Arrange a musical score for a smaller ensemble
- All notes are taken from the original score
- Each instrument can only play one note at a time

Joseph Haydn playing in a string quartet,
painting from the StaatsMuseum, Vienna



- What are we trying to do? What are the constraints to our problem?
- Take a musical score and reduce it to a smaller ensemble
- All notes must be taken from the original score, no new notes can be added
- Each instrument can only take notes from one part at a time

Method

- Formulating arrangement as a problem to be solved via annealing, five-step process
- Split parts into musical phrases
- Arrange phrases into a graph (will explain later)
- Formulate the optimisation problem
- Solve corresponding graph problem using a quantum computer
- Construct final arrangement from the solution returned

Method

1. Split score into musical phrases

Method
1. Split score into musical phrases

- Formulating arrangement as a problem to be solved via annealing, five-step process
- Split parts into musical phrases
- Arrange phrases into a graph (will explain later)
- Formulate the optimisation problem
- Solve corresponding graph problem using a quantum computer
- Construct final arrangement from the solution returned

Method

1. Split score into musical phrases
2. Arrange phrases into a graph

Method
1. Split score into musical phrases
2. Arrange phrases into a graph

- Formulating arrangement as a problem to be solved via annealing, five-step process
- Split parts into musical phrases
- Arrange phrases into a graph (will explain later)
- Formulate the optimisation problem
- Solve corresponding graph problem using a quantum computer
- Construct final arrangement from the solution returned

Method

1. Split score into musical phrases
2. Arrange phrases into a graph
3. Formulate optimisation problem

Method
1. Split score into musical phrases
2. Arrange phrases into a graph
3. Formulate optimisation problem

- Formulating arrangement as a problem to be solved via annealing, five-step process
- Split parts into musical phrases
- Arrange phrases into a graph (will explain later)
- Formulate the optimisation problem
- Solve corresponding graph problem using a quantum computer
- Construct final arrangement from the solution returned

Method

1. Split score into musical phrases
2. Arrange phrases into a graph
3. Formulate optimisation problem
4. Solve problem using QPU

Method

1. Split score into musical phrases
2. Arrange phrases into a graph
3. Formulate optimisation problem
4. Solve problem using QPU

- Formulating arrangement as a problem to be solved via annealing, five-step process
- Split parts into musical phrases
- Arrange phrases into a graph (will explain later)
- Formulate the optimisation problem
- Solve corresponding graph problem using a quantum computer
- Construct final arrangement from the solution returned

Method

1. Split score into musical phrases
2. Arrange phrases into a graph
3. Formulate optimisation problem
4. Solve problem using QPU
5. Construct arrangement from solution

Method
1. Split score into musical phrases
2. Arrange phrases into a graph
3. Formulate optimisation problem
4. Solve problem using QPU
5. Construct arrangement from solution

- Formulating arrangement as a problem to be solved via annealing, five-step process
- Split parts into musical phrases
- Arrange phrases into a graph (will explain later)
- Formulate the optimisation problem
- Solve corresponding graph problem using a quantum computer
- Construct final arrangement from the solution returned

1. Split score

[6]Cambouropoulos, 'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.

- First stage to separate each part of original score into phrases
- Phrases — smallest unit of music that preserves melody and structure
- Boundaries between phrases found using LBDM
- Measures the degree of change of a certain parameter (x) between notes (i) (explain equation)
- Strength calculated for both pitch and IOI, weighted and summed to give the final strength
- Strengths above a threshold value are considered phrases

1. Split score

- Musical phrases chosen as smallest unit of music
- Preserve melody and structure when rearranged

[6] Cambouropoulos, ‘The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing’.

1. Split score
- Musical phrases chosen as smallest unit of music
- Preserve melody and structure when rearranged
6Cambouropoulos, ‘The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing’.

- First stage to separate each part of original score into phrases
- Phrases — smallest unit of music that preserves melody and structure
- Boundaries between phrases found using LBDM
- Measures the degree of change of a certain parameter (x) between notes (i) (explain equation)
- Strength calculated for both pitch and IOI, weighted and summed to give the final strength
- Strengths above a threshold value are considered phrases

1. Split score

- Musical phrases chosen as smallest unit of music
- Preserve melody and structure when rearranged

Local boundary detection model (LBDM)[6]

$$S_i = x_i \times (r_{i-1,i} + r_{i,i+1})$$

[6] Cambouropoulos, 'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.

1. Split score
- Musical phrases chosen as smallest unit of music
- Preserve melody and structure when rearranged
Local boundary detection model (LBDM)[6]
$$S_i = x_i \times (r_{i-1,i} + r_{i,i+1})$$
[6] Cambouropoulos, 'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.

- First stage to separate each part of original score into phrases
- Phrases — smallest unit of music that preserves melody and structure
- Boundaries between phrases found using LBDM
- Measures the degree of change of a certain parameter (x) between notes (i) (explain equation)
- Strength calculated for both pitch and IOI, weighted and summed to give the final strength
- Strengths above a threshold value are considered phrases

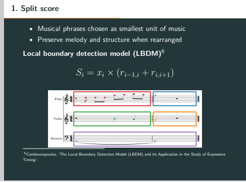
1. Split score

- Musical phrases chosen as smallest unit of music
- Preserve melody and structure when rearranged

Local boundary detection model (LBDM)[6]

$$S_i = x_i \times (r_{i-1,i} + r_{i,i+1})$$

[6] Cambouropoulos, 'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.



- First stage to separate each part of original score into phrases
- Phrases — smallest unit of music that preserves melody and structure
- Boundaries between phrases found using LBDM
- Measures the degree of change of a certain parameter (x) between notes (i) (explain equation)
- Strength calculated for both pitch and IOI, weighted and summed to give the final strength
- Strengths above a threshold value are considered phrases

2. Create graph

- What is a graph? Nodes connected by edges, useful to model pairwise relations between objects
- Each phrase becomes a node, edges between nodes if phrases overlap (play at the same time)

2. Create graph

2. Create graph

- Vertices (nodes) connected by edges
- Models pairwise relations between objects

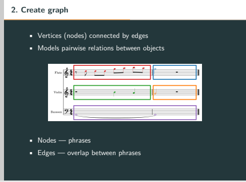
2. Create graph
- Vertices (nodes) connected by edges
- Models pairwise relations between objects

- What is a graph? Nodes connected by edges, useful to model pairwise relations between objects
- Each phrase becomes a node, edges between nodes if phrases overlap (play at the same time)

2. Create graph

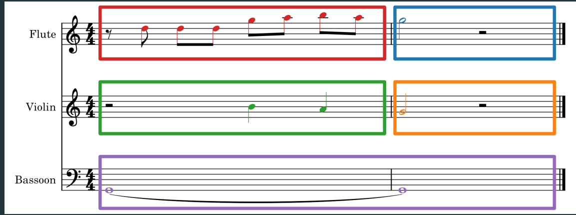
- Vertices (nodes) connected by edges
- Models pairwise relations between objects

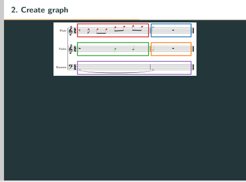
- Nodes — phrases
- Edges — overlap between phrases



- What is a graph? Nodes connected by edges, useful to model pairwise relations between objects
- Each phrase becomes a node, edges between nodes if phrases overlap (play at the same time)

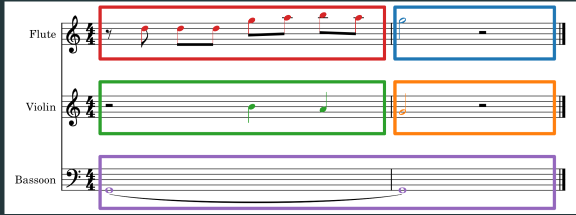
2. Create graph

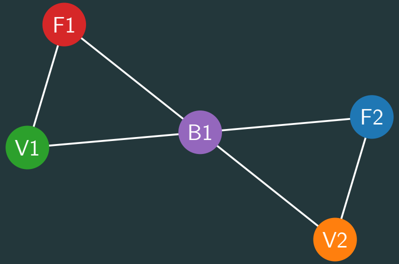


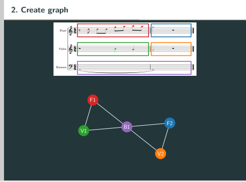


- Score on top becomes graph on bottom

2. Create graph







- Score on top becomes graph on bottom

3. Create optimisation problem

3. Create optimisation problem

- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

3. Create optimisation problem

Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

3. Create optimisation problem
Proper vertex colouring
Colour each vertex such that no edge connects two vertices of the same colour

- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

3. Create optimisation problem

Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

3. Create optimisation problem

Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

3. Create optimisation problem

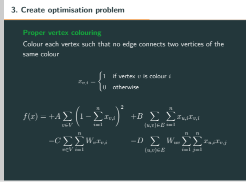
Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

$$f(x) = +A\sum_{v\in V}\left(1 - \sum_{i=1}^{n} x_{v,i}\right)^2 \quad +B\sum_{(u,v)\in E}\sum_{i=1}^{n} x_{u,i}x_{v,i}$$

$$-C\sum_{v\in V}\sum_{i=1}^{n} W_v x_{v,i} \qquad -D\sum_{(u,v)\in E} W_{uv}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{u,i}x_{v,j}$$



- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

3. Create optimisation problem

Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned} f(x) = &+A \sum_{v \in V} \left(1 - \sum_{i=1}^{n} x_{v,i}\right)^2 &&+B \sum_{(u,v)\in E} \sum_{i=1}^{n} x_{u,i} x_{v,i} \\ &-C \sum_{v \in V} \sum_{i=1}^{n} W_v x_{v,i} &&-D \sum_{(u,v)\in E} W_{uv} \sum_{i=1}^{n} \sum_{j=1}^{n} x_{u,i} x_{v,j} \end{aligned}$$

3. Create optimisation problem
Proper vertex colouring
Colour each vertex such that no edge connects two vertices of the same colour

- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

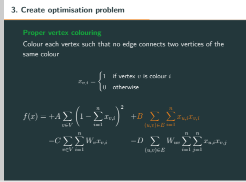
3. Create optimisation problem

Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned} f(x) = &+A \sum_{v \in V} \left(1 - \sum_{i=1}^{n} x_{v,i}\right)^2 \quad +B \sum_{(u,v)\in E} \sum_{i=1}^{n} x_{u,i} x_{v,i} \\ &-C \sum_{v \in V} \sum_{i=1}^{n} W_v x_{v,i} \quad -D \sum_{(u,v)\in E} W_{uv} \sum_{i=1}^{n} \sum_{j=1}^{n} x_{u,i} x_{v,j} \end{aligned}$$



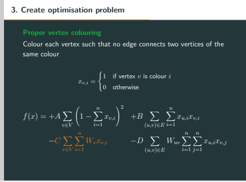
- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

Proper vertex colouring

Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned} f(x) = &+A \sum_{v \in V} \left(1 - \sum_{i=1}^{n} x_{v,i}\right)^2 &&+B \sum_{(u,v) \in E} \sum_{i=1}^{n} x_{u,i} x_{v,i} \\ &-C \sum_{v \in V} \sum_{i=1}^{n} W_v x_{v,i} &&-D \sum_{(u,v) \in E} W_{uv} \sum_{i=1}^{n} \sum_{j=1}^{n} x_{u,i} x_{v,j} \end{aligned}$$



- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

3. Create optimisation problem

Proper vertex colouring

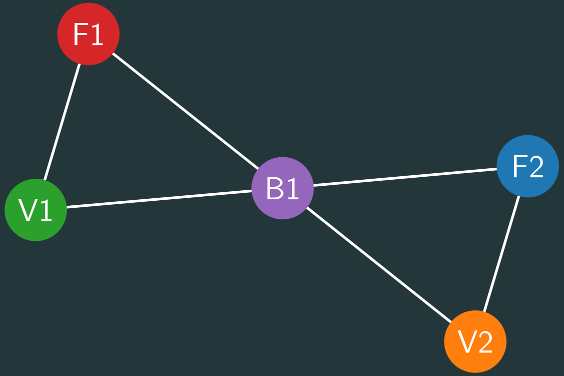
Colour each vertex such that no edge connects two vertices of the same colour

$$x_{v,i} = \begin{cases} 1 & \text{if vertex } v \text{ is colour } i \\ 0 & \text{otherwise} \end{cases}$$

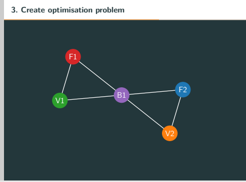
$$\begin{aligned} f(x) = &+A \sum_{v\in V}\left(1-\sum_{i=1}^{n} x_{v,i}\right)^2 &&+B \sum_{(u,v)\in E}\sum_{i=1}^{n} x_{u,i}x_{v,i} \\ &-C \sum_{v\in V}\sum_{i=1}^{n} W_v x_{v,i} &&-D \sum_{(u,v)\in E} W_{uv} \sum_{i=1}^{n}\sum_{j=1}^{n} x_{u,i}x_{v,j} \end{aligned}$$

- Use a graph theory problem to create the optimisation problem that matches our constraints
- Here each colour represents an instrument we are arranging for
- QUBO, set of n colours, $x_{v,i}$ is 1 if node v is colour i
- A — each node is only coloured once, sum over colours is one
- B — penalise adjacent nodes with the same colour
- C — weight of each node, preference for certain nodes
- D — weight of each edge, preference for certain edges
- Weights here are musical entropy i.e. how interesting the phrase is musically

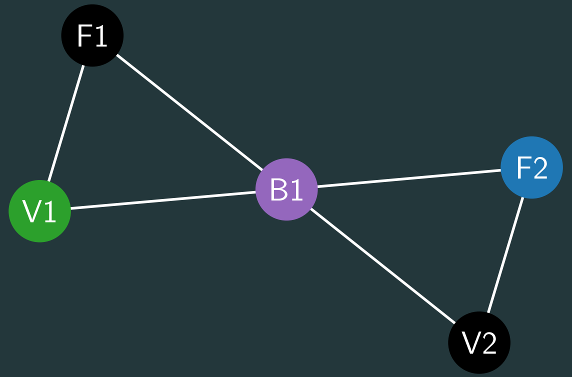
3. Create optimisation problem

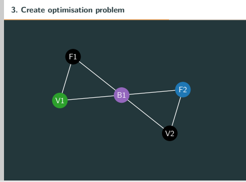


2025-03-09



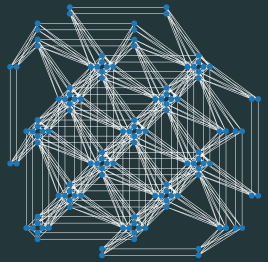
3. Create optimisation problem



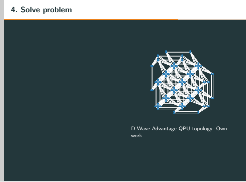


- $n = 1$
- One of many possible solutions

4. Solve problem



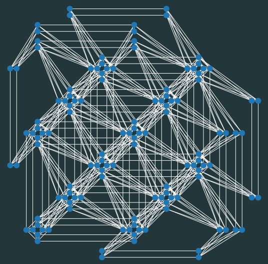
D-Wave Advantage QPU topology. Own work.



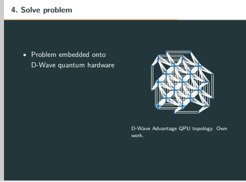
- D-Wave Systems is a company that gives access to true quantum annealers, normally for business applications
- Interact via a Python SDK, submit problems to the QPU
- Returns a distribution of results, each with an associated energy
- Run the problem thousands of times to find the lowest-energy solutions

4. Solve problem

- Problem embedded onto D-Wave quantum hardware



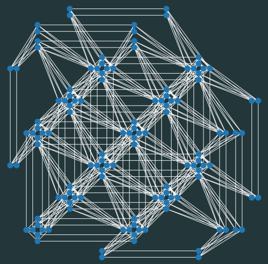
D-Wave Advantage QPU topology. Own work.



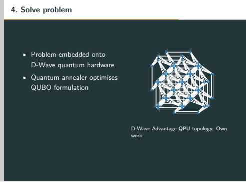
- D-Wave Systems is a company that gives access to true quantum annealers, normally for business applications
- Interact via a Python SDK, submit problems to the QPU
- Returns a distribution of results, each with an associated energy
- Run the problem thousands of times to find the lowest-energy solutions

4. Solve problem

- Problem embedded onto D-Wave quantum hardware
- Quantum annealer optimises QUBO formulation



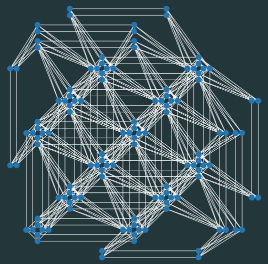
D-Wave Advantage QPU topology. Own work.



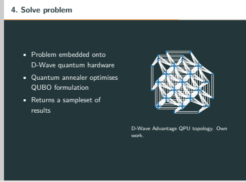
- D-Wave Systems is a company that gives access to true quantum annealers, normally for business applications
- Interact via a Python SDK, submit problems to the QPU
- Returns a distribution of results, each with an associated energy
- Run the problem thousands of times to find the lowest-energy solutions

4. Solve problem

- Problem embedded onto D-Wave quantum hardware
- Quantum annealer optimises QUBO formulation
- Returns a sampleset of results



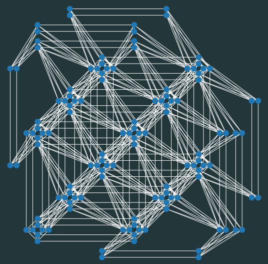
D-Wave Advantage QPU topology. Own work.



- D-Wave Systems is a company that gives access to true quantum annealers, normally for business applications
- Interact via a Python SDK, submit problems to the QPU
- Returns a distribution of results, each with an associated energy
- Run the problem thousands of times to find the lowest-energy solutions

4. Solve problem

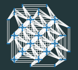
- Problem embedded onto D-Wave quantum hardware
- Quantum annealer optimises QUBO formulation
- Returns a sampleset of results
- Run many times to find lowest-energy solution



D-Wave Advantage QPU topology. Own work.

4. Solve problem

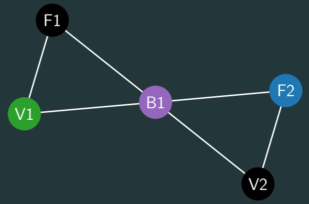
- Problem embedded onto D-Wave quantum hardware
- Quantum annealer optimises QUBO formulation
- Returns a sampleset of results
- Run many times to find lowest-energy solution

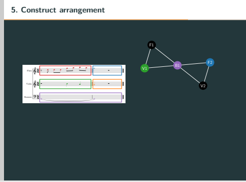


D-Wave Advantage QPU topology. Own work.

- D-Wave Systems is a company that gives access to true quantum annealers, normally for business applications
- Interact via a Python SDK, submit problems to the QPU
- Returns a distribution of results, each with an associated energy
- Run the problem thousands of times to find the lowest-energy solutions

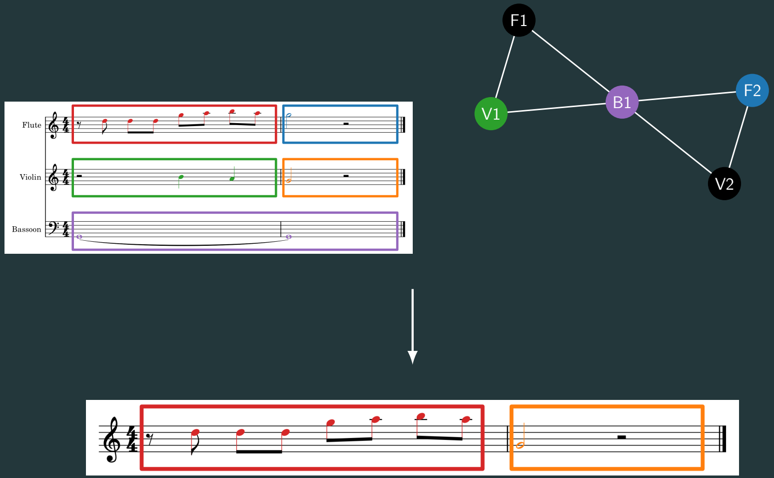
5. Construct arrangement

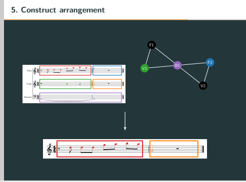




- Take chosen low-energy solution and construct the final arrangement
- Map each node back to its phrase, with colour corresponding to the instrument

5. Construct arrangement





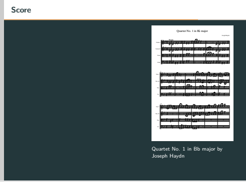
- Take chosen low-energy solution and construct the final arrangement
- Map each node back to its phrase, with colour corresponding to the instrument

Results

Results

Score

Quartet No. 1 in Bb major by Joseph Haydn

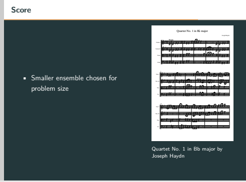


- Quartet No. 1 in Bb major by Joseph Haydn
- Smaller instrumentation and length (about 3 min), keeping the problem graph small and manageable
- Musical style has well-defined structure and phrases

Score

- Smaller ensemble chosen for problem size

Quartet No. 1 in Bb major by Joseph Haydn

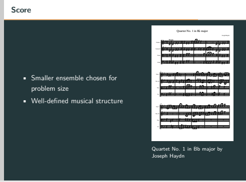


- Quartet No. 1 in Bb major by Joseph Haydn
- Smaller instrumentation and length (about 3 min), keeping the problem graph small and manageable
- Musical style has well-defined structure and phrases

Score

- Smaller ensemble chosen for problem size
- Well-defined musical structure

Quartet No. 1 in Bb major by Joseph Haydn

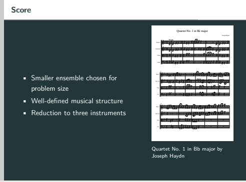


- Quartet No. 1 in Bb major by Joseph Haydn
- Smaller instrumentation and length (about 3 min), keeping the problem graph small and manageable
- Musical style has well-defined structure and phrases

Score

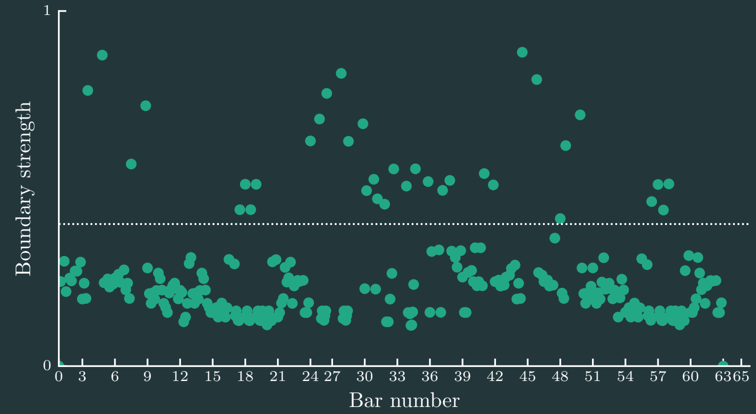
- Smaller ensemble chosen for problem size
- Well-defined musical structure
- Reduction to three instruments

Quartet No. 1 in Bb major by Joseph Haydn

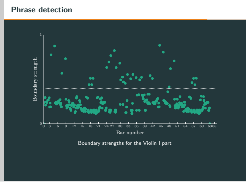


- Quartet No. 1 in Bb major by Joseph Haydn
- Smaller instrumentation and length (about 3 min), keeping the problem graph small and manageable
- Musical style has well-defined structure and phrases

Phrase detection

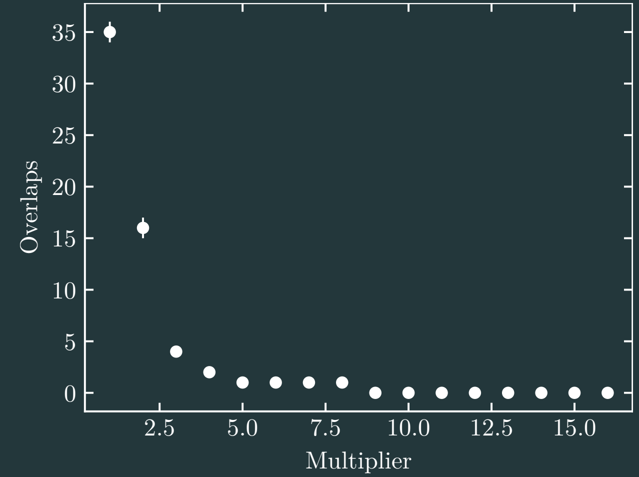


Boundary strengths for the Violin I part

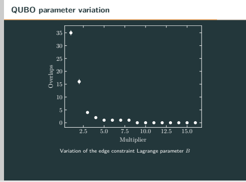


- Example of the LBDM finding suitable boundaries for phrases
- Threshold value of 0.4 chosen manually

QUBO parameter variation

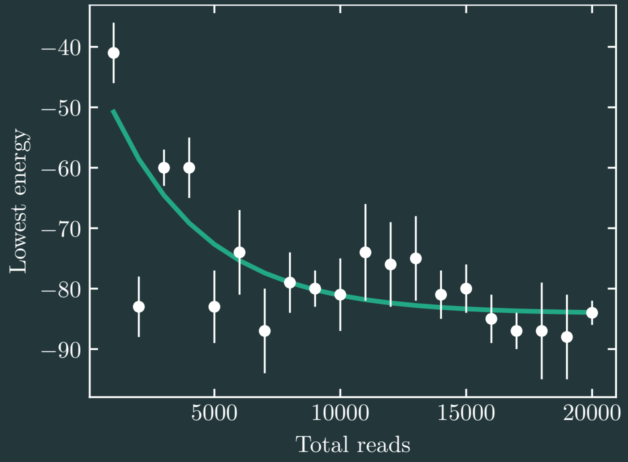


Variation of the edge constraint Lagrange parameter B

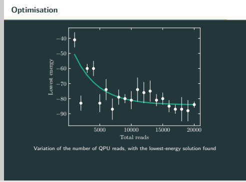


- Each QUBO problem submitted five times with different edge constraint Lagrange parameter
- Checking against fulfillment of the desired constraint
- Lagrange parameters taken as multipliers of the maximum node weight for normalisation
- 12.0 chosen as the best parameter, with all others equal to one

Optimisation



Variation of the number of QPU reads, with the lowest-energy solution found



- Once Lagrange parameters chosen, can check how well the annealer optmises the problem
- In general, more reads is more likely to find lower-energy solutions
- Sometimes the annealer gets lucky (see 2000 reads)
- Each number of reads repeated five times, exponential decay fitted

Conclusions

Conclusions

Conclusions

- Successful application of this method on a new problem
- QPU returns samples that fulfill the constraints of the problem, creating a valid arrangement
- New technology, limited in power
- What would it take for quantum to show advantage?

Conclusions

- Successful novel application of quantum annealing

Conclusions
- Successful novel application of quantum annealing

- Successful application of this method on a new problem
- QPU returns samples that fulfill the constraints of the problem, creating a valid arrangement
- New technology, limited in power
- What would it take for quantum to show advantage?

Conclusions

- Successful novel application of quantum annealing
- QPU returns low-energy samples

Conclusions
- Successful novel application of quantum annealing
- QPU returns low-energy samples

- Successful application of this method on a new problem
- QPU returns samples that fulfill the constraints of the problem, creating a valid arrangement
- New technology, limited in power
- What would it take for quantum to show advantage?

Conclusions

- Successful novel application of quantum annealing
- QPU returns low-energy samples
- Necessary constraints for a valid arrangement fulfilled

Conclusions
- Successful novel application of quantum annealing
- QPU returns low-energy samples
- Necessary constraints for a valid arrangement fulfilled

- Successful application of this method on a new problem
- QPU returns samples that fulfill the constraints of the problem, creating a valid arrangement
- New technology, limited in power
- What would it take for quantum to show advantage?

Conclusions

- Successful novel application of quantum annealing
- QPU returns low-energy samples
- Necessary constraints for a valid arrangement fulfilled
- Still very new technology, does not show quantum advantage (yet)

Conclusions

- Successful novel application of quantum annealing
- QPU returns low-energy samples
- Necessary constraints for a valid arrangement fulfilled
- Still very new technology, does not show quantum advantage (yet)

- Successful application of this method on a new problem
- QPU returns samples that fulfill the constraints of the problem, creating a valid arrangement
- New technology, limited in power
- What would it take for quantum to show advantage?

Future work

[7] Pearce and Wiggins, 'Towards A Framework for the Evaluation of Machine Compositions'.

- How well does the method scale with larger scores? How well can it find low energies with smaller problems?
- Compare to classical optimisation methods, time to solution, energy of solutions
- Only tuned one parameter by hand, could use a more systematic approach to find lower-energy solutions
- Quality judgement — Turing-like test, present subjects with human-/computer-generated scores

Future work

- Variation in problem size

[7] Pearce and Wiggins, ‘Towards A Framework for the Evaluation of Machine Compositions’.

- How well does the method scale with larger scores? How well can it find low energies with smaller problems?
- Compare to classical optimisation methods, time to solution, energy of solutions
- Only tuned one parameter by hand, could use a more systematic approach to find lower-energy solutions
- Quality judgement — Turing-like test, present subjects with human-/computer-generated scores

Future work

- Variation in problem size
- Comparison to classical methods

[7] Pearce and Wiggins, 'Towards A Framework for the Evaluation of Machine Compositions'.

Future work

- Variation in problem size
- Comparison to classical methods

[7] Pearce and Wiggins, 'Towards A Framework for the Evaluation of Machine Compositions'.

- How well does the method scale with larger scores? How well can it find low energies with smaller problems?
- Compare to classical optimisation methods, time to solution, energy of solutions
- Only tuned one parameter by hand, could use a more systematic approach to find lower-energy solutions
- Quality judgement — Turing-like test, present subjects with human-/computer-generated scores

Future work

- Variation in problem size
- Comparison to classical methods
- Lagrange parameter tuning

[7] Pearce and Wiggins, ‘Towards A Framework for the Evaluation of Machine Compositions’.

Future work
- Variation in problem size
- Comparison to classical methods
- Lagrange parameter tuning
[7] Pearce and Wiggins, ‘Towards A Framework for the Evaluation of Machine Compositions’.

- How well does the method scale with larger scores? How well can it find low energies with smaller problems?
- Compare to classical optimisation methods, time to solution, energy of solutions
- Only tuned one parameter by hand, could use a more systematic approach to find lower-energy solutions
- Quality judgement — Turing-like test, present subjects with human-/computer-generated scores

Future work

- Variation in problem size
- Comparison to classical methods
- Lagrange parameter tuning
- Qualitative judgement of computer arrangements[7]

[7] Pearce and Wiggins, ‘Towards A Framework for the Evaluation of Machine Compositions’.

Future work
- Variation in problem size
- Comparison to classical methods
- Lagrange parameter tuning
- Qualitative judgement of computer arrangements[7]
[7] Pearce and Wiggins, ‘Towards A Framework for the Evaluation of Machine Compositions’.

- How well does the method scale with larger scores? How well can it find low energies with smaller problems?
- Compare to classical optimisation methods, time to solution, energy of solutions
- Only tuned one parameter by hand, could use a more systematic approach to find lower-energy solutions
- Quality judgement — Turing-like test, present subjects with human-/computer-generated scores

Thank you!

Thank you!

Quantum Annealing for Music Arrangement

Lucas Kirby

4 March 2025

Department of Physics, Durham University

2025-03-09

Quantum Annealing for Music Arrangement

Quantum Annealing
for Music Arrangement

Lucas Kirby
4 March 2025
Department of Physics, Durham University

- Arya, Ashish et al. **'Music Composition Using Quantum Annealing'.** In: *arXiv* (Jan. 2022). DOI: 10.48550/arXiv.2201.10557. (Visited on 26/10/2024).
- Born, M. and V. Fock. **'Beweis des Adiabatensatzes'.** de. In: *Zeitschrift für Physik* 51.3 (Mar. 1928), pp. 165–180. ISSN: 0044-3328. DOI: 10.1007/BF01343193. URL: https://doi.org/10.1007/BF01343193 (visited on 01/03/2025).
- Cambouropoulos, Emilios. **'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.** In: *International Computer Music Association* (2011). ISSN: 2223-3881.

2025-03-09

Quantum Annealing for Music Arrangement

└References

References i

- Arya, Ashish et al. **'Music Composition Using Quantum Annealing'.** In: *arXiv* (Jan. 2022). DOI: 10.48550/arXiv.2201.10557. (Visited on 26/10/2024).
- Born, M. and V. Fock. **'Beweis des Adiabatensatzes'.** de. In: *Zeitschrift für Physik* 51.3 (Mar. 1928), pp. 165–180. ISSN: 0044-3328. DOI: 10.1007/BF01343193. URL: https://doi.org/10.1007/BF01343193 (visited on 01/03/2025).
- Cambouropoulos, Emilios. **'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.** In: *International Computer Music Association* (2011). ISSN: 2223-3881.

References ii

- Freedline, Alex. **'Algorhythms: Generating Music with D-Wave's Quantum Annealer'.** en. In: *MIT 6.s089—Intro to Quantum Computing* (Feb. 2021).
- Huang, Jiun-Long, Shih-Chuan Chiu and Man-Kwan Shan. **'Towards an automatic music arrangement framework using score reduction'.** In: *ACM Trans. Multimedia Comput. Commun. Appl.* 8.1 (Feb. 2012), 8:1–8:23. ISSN: 1551-6857. DOI: 10.1145/2071396.2071404. (Visited on 05/12/2024).

References ii

- Freedline, Alex. **'Algorhythms: Generating Music with D-Wave's Quantum Annealer'.** en. In: *MIT 6.s089—Intro to Quantum Computing* (Feb. 2021).
- Huang, Jiun-Long, Shih-Chuan Chiu and Man-Kwan Shan. **'Towards an automatic music arrangement framework using score reduction'.** In: *ACM Trans. Multimedia Comput. Commun. Appl.* 8.1 (Feb. 2012), 8:1–8:23. ISSN: 1551-6857. DOI: 10.1145/2071396.2071404. (Visited on 05/12/2024).

- Li, You et al. **‘Automatic Piano Reduction of Orchestral Music Based on Musical Entropy’.** In: *2019 53rd Annual Conference on Information Sciences and Systems (CISS)*. Mar. 2019, pp. 1–5. DOI: 10.1109/CISS.2019.8693036. URL: https://ieeexplore.ieee.org/document/8693036 (visited on 27/12/2024).
- Miranda, Eduardo Reck, ed. ***Quantum Computer Music: Foundations, Methods and Advanced Concepts.*** en. Springer International Publishing, 2022. ISBN: 978-3-031-13908-6 978-3-031-13909-3. DOI: 10.1007/978-3-031-13909-3. (Visited on 28/12/2024).

References iii

Li, You et al. ‘Automatic Piano Reduction of Orchestral Music Based on Musical Entropy’. In: 2019 53rd Annual Conference on Information Sciences and Systems (CISS). Mar. 2019, pp. 1–5. DOI: 10.1109/CISS.2019.8693036. URL: https://ieeexplore.ieee.org/document/8693036 (visited on 27/12/2024).

Miranda, Eduardo Reck, ed. Quantum Computer Music: Foundations, Methods and Advanced Concepts. en. Springer International Publishing, 2022. ISBN: 978-3-031-13908-6 978-3-031-13909-3. DOI: 10.1007/978-3-031-13909-3. (Visited on 28/12/2024).

References iv

- Moses, William S. and Erik D. Demaine. **‘Computational Complexity of Arranging Music’.** In: *arXiv* (July 2016). arXiv:1607.04220. DOI: `10.48550/arXiv.1607.04220`. (Visited on 09/11/2024).
- Nakamura, Eita and Kazuyoshi Yoshii. **‘Statistical piano reduction controlling performance difficulty’.** en. In: *APSIPA Transactions on Signal and Information Processing* 7 (Jan. 2018), e13. ISSN: 2048-7703. DOI: `10.1017/ATSIP.2018.18`. (Visited on 17/12/2024).
- Pearce, M. and Geraint A. Wiggins. **‘Towards A Framework for the Evaluation of Machine Compositions’.** In: *Proceedings of the AISB’01 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*. 2001.

References iv

- Moses, William S. and Erik D. Demaine. **‘Computational Complexity of Arranging Music’.** In: *arXiv* (July 2016). arXiv:1607.04220. DOI: `10.48550/arXiv.1607.04220`. (Visited on 09/11/2024).
- Nakamura, Eita and Kazuyoshi Yoshii. **‘Statistical piano reduction controlling performance difficulty’.** en. In: *APSIPA Transactions on Signal and Information Processing* 7 (Jan. 2018), e13. ISSN: 2048-7703. DOI: `10.1017/ATSIP.2018.18`. (Visited on 17/12/2024).
- Pearce, M. and Geraint A. Wiggins. **‘Towards A Framework for the Evaluation of Machine Compositions’.** In: *Proceedings of the AISB’01 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*. 2001.

LBDM

Boundary strength

$$S_i = x_i \times (r_{i-1,i} + r_{i,i+1})$$
$$r_{i,i+1} = \frac{|x_i - x_{i+1}|}{x_i + x_{i+1}}$$

Normalisation

$$S_i' = \frac{S_i - \min(S_i)}{\max(S_i) - \min(S_i)}$$

Weighting

$$S = \frac{1}{3}\left(S'_{\text{pitch}} + 2S'_{\text{IOI}}\right)$$

[8]

[8]Cambouropoulos, 'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.

LBDM
Boundary strength
$$S_i = x_i \times (r_{i-1,i} + r_{i,i+1})$$
$$r_{i,i+1} = \frac{|x_i - x_{i+1}|}{x_i + x_{i+1}}$$
Normalisation
$$S_i' = \frac{S_i - \min(S_i)}{\max(S_i) - \min(S_i)}$$
Weighting
$$S = \frac{1}{3}\left(S'_{\text{pitch}} + 2S'_{\text{IOI}}\right)$$
8
[8]Cambouropoulos, 'The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing'.

- Boundaries always taken at beginning/end of piece
- Weightings derived by trial and error

Phrase entropy

x_i — parameter x of note i

Shannon entropy

$$H(X) := -\sum_i P(x_i)\log_2 P(x_i)$$

Probability distribution

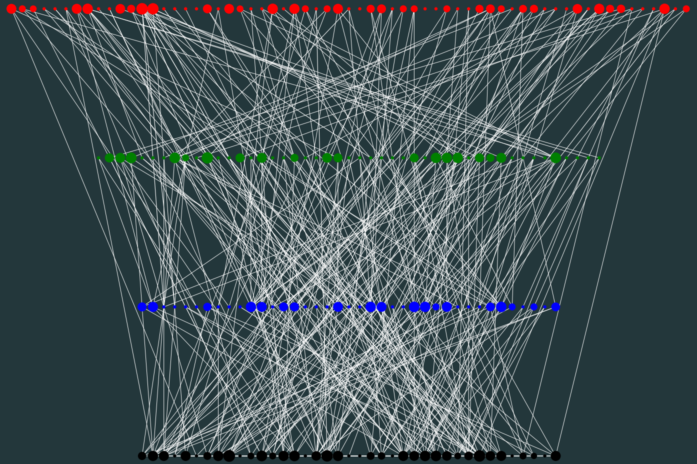
$$P(x_i) = \frac{n_i}{N}$$

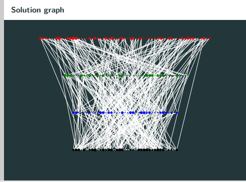
[9] Li et al., 'Automatic Piano Reduction of Orchestral Music Based on Musical Entropy'.

Phrase entropy
x_i — parameter x of note i
Shannon entropy
$H(X) := -\sum_i P(x_i)\log_2 P(x_i)$
Probability distribution
$P(x_i) = \frac{n_i}{N}$
9
[9] Li et al., 'Automatic Piano Reduction of Orchestral Music Based on Musical Entropy'.

- Shannon entropy units in bits due to $\log_2$
- Distribution calculated for pitch and duration

Solution graph





Solution score