EMBnet.journal
Bioinformatics in Action

qualign: solving sequence alignment based on quadratic unconstrained binary optimisation

Yuki Matsumoto✉, **Shota Nakamura**

Research Institute for Microbial Diseases, Osaka University, Osaka, Japan
Competing interests: YM none; SN none

Abstract

Bioinformatics has, among others, the issue of solving complex computational problems with vast amounts of sequencing data. Recently, a new computing architecture, the annealing machine, has emerged that applies to actual problems and is available for practical use. This novel architecture can solve discrete optimisation problems by replacing algorithms designed under the von Neumann architecture. To perform computations on the annealing machine, quadratic unconstrained binary optimisation (QUBO) formulations should be constructed and optimised according to the application. In this study, we developed an algorithm under the annealing machine architecture to solve sequence alignment problems, a known fundamental process widely used in genetic analysis, such as mutation detection and genome assembly. We constructed a QUBO formulation based on dynamic programming to solve a pairwise sequence alignment and derived its general form. We compared with conventional methods to solve 40 bp of pairwise alignment problem. Our implementation, named qualign, solved sequence alignment problems with accuracy comparable to that of conventional methods. Although a small pairwise alignment was solved owing to the limited memory size of this method, this is the first step of the application of annealing machines. We showed that our QUBO formulation solved the sequencing alignment problem. In the future, increasing the memory size of annealing machine will allow annealing machines to impact a wide range of bioinformatics applications positively.
Availability: the source code of qualign is available at https://github.com/ymatsumoto/qualign

Introduction

In bioinformatics, there are often discrete optimisation problems belonging to the NP-Hard or NP-complete class, such as sequence alignment, genome sequence assembly, and structural estimation (Wang and Jiang, 1994; Medvedev *et al.*, 2007; Crescenzi *et al.*, 1998). Current techniques are effective in solving such small-scale problems of aligning short sequences, but not large ones requiring finding all overlaps among vast sequences obtained from a huge genome. Recently, the amount of sequencing data generated using actively developing next-generation sequencing technologies is growing faster than Moore's law, an exponential growth of a dense integrated circuit over time (Mardis, 2011).

Annealing machines have emerged as a new computing paradigm and have become readily available for practical use with quantum (Jünger *et al.*, 2021) and complementary metal-oxide-semiconductor (CMOS)-implemented hardware (Boixo *et al.*, 2014; Yoshimura *et al.*, 2020; Aramon *et al.*, 2019). In particular, quantum annealing machines are one step ahead of general-purpose quantum computers because of their computation using quantum effects; moreover, they are expected to solve discrete optimisation problems [6, 9] efficiently. The various algorithms designed for von Neumann architecture need to be theoretically converted into a quadratic unconstrained binary optimisation (QUBO) formulation because annealing machines require a QUBO model as an input for computation (Lucas, 2014).

Local sequence alignment is one of the most fundamental processes in bioinformatics (Smith and Waterman, 1981). Although the exact solution to this problem can be obtained using dynamic programming, this process belongs to the NP-Hard class (Wang and Jiang, 1994). In this study, we developed a novel sequence alignment algorithm using QUBO formulation based on dynamic programming, named qualign, derived from QUbo-based sequence ALIGNment, to solve a local sequence alignment using an annealing machine.

Article history
Received: 02 May 2022
Accepted: 28 June 2022
Published: 08 March 2023

Matsumoto et al. (2022) EMBnet.journal 28, e1020
http://dx.doi.org/10.14806/ej.28.0.1020

Materials, Methodologies and Techniques

We constructed a QUBO formulation to solve sequence alignment problems using the annealing machine as equation (1) represented by a Hamiltonian form. This equation is composed of three major terms. The first term assesses whether the alignment is matched or not according to the scoring of the BLOSUM62 matrix constant (Eddy, 2004). The second term represents the matching character between two input sequences and limits the number of aligned characters to one at most. The last term limits the occurrence of base swaps before and after a base when it is aligned.

$$\begin{aligned} H = C_0 \sum_{\substack{0\le i<N \\ 0\le j<N}} A_{s_i^1 s_j^2} x_{i,j} + C_1 \left(\sum_{0\le j<N} \left(1 - \sum_{0\le i<N} x_{i,j}\right)^2 + \sum_{0\le i<N} \left(1 - \sum_{0\le j<N} x_{i,j}\right)^2 \right) \\ + C_2 \sum_{\substack{0\le i<N \\ 0\le j<N}} x_{i,j} \left(\left(\sum_{\substack{0\le i<N \\ 0\le j<N}} x_{a,b} \right) - \left(\sum_{a\le i \wedge b\le j} x_{a,b} \right) - \left(\sum_{i\le a \wedge j\le b} x_{a,b} \right) + x_{i,j} \right) \end{aligned} \tag{1}$$

Here, N is the number of characters in an input sequence. xi,j represents whether i- and j-th characters were aligned, and their values were allocated to each qubit as the Boolean values. C_0, C_1, and C_2 represent the weight coefficient of each term. s_i^1 and s_j^2 represent the pairwise sequence alignment of the inputs. A represents the alignment-score matrix that returns a matching score between s_i^1 and s_j^2.

To formulate the sum of the products, the two input sequences were initially taken into the computer memory and converted to a QUBO formulation using the pyQUBO package (Figure 1). To solve the sequence alignment problem using the QUBO model, we designed qualign to set each of the coefficient variables in equation (1) to C0=4, C1=8, and C2=8 as the default setting. The converted QUBO model was solved using an actual annealing machine or simulated annealing sampler, dwave-neal provided by D-wave, the vendor developing the quantum annealing machine. Our design currently supports three kinds of solvers, including the D-wave Quantum annealing machine, the Fujitsu Digital annealer, and the dwave-neal solver.

Results

We evaluated the alignment accuracy of the qualign against other conventional methods when asked to align pair-wised sequences selected from the benchmark alignment database, BAliBASE (Thompson *et al.*, 1999) and trimmed the sequence length to 40 bp. The resulting alignment score using the qualign was -17, which improved from -30 without alignment. The alignment scores of ClustalW (Thompson *et al.*, 1994) and MUSCLE (Edgar, 2004) were -17 and -140, respectively (see Supplementary data file). These results indicate that the accuracy of qualign was comparable to the conventional software tools.

We developed a novel local sequence alignment algorithm based on a QUBO model using an annealing machine. Our model could be generalised to a multiple-sequence alignment with an arbitrary number of sequences. The general form of equation (1) is represented by equation (2), where i and j represent a set of indices and an element of I, the set of all combinations of indices, and the function ei(k) returns the k-th element of i.

$$\begin{aligned} H = C_0 \sum_{i\in I} A_{S(i)} x_i + C_1 \sum_{i\in I} \sum_{0\le m<M} x_i \left(1 - \sum_{\{j\in I \mid 0\le i(m)<N\}} x_j \right)^2 \\ + C_2 \sum_{i \in I} x_i \left(x_i + \sum_{j\in I} x_j - \sum_{\{j\in I \mid \forall k\, e_i(k)\le e_j(k)\}} x_j - \sum_{\{j\in I \mid \forall k\, e_j(k)\le e_i(k)\}} x_j \right) \end{aligned} \tag{2}$$

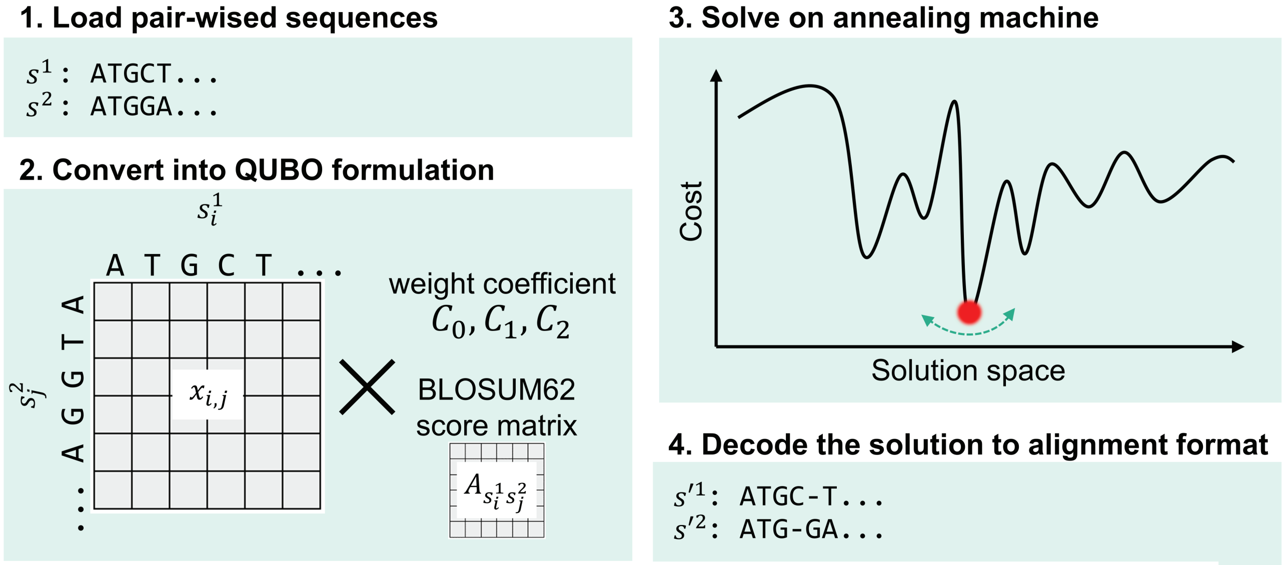

Figure 1. Algorithm workflow of qualign. The computation on the annealing machine is performed in step 3, steps 1, 2, and 4 are performed on a conventional computer.

Although the annealing machine could solve the problem in a given time, it required a QUBO model as the input. In the current implementation of qualign, the number of input sequences was limited to two due to restriction of the memory size of the annealing machines (Boixo *et al.*, 2014; Aramon *et al.*, 2019).

Conclusions

We showed that the proposed algorithm using QUBO model solved the sequence alignment problem using the new computational paradigm under annealing machines. The accurate sequence alignment could also lead to optimised results of genome assembly or mutation detection because sequence alignment was performed in the initial steps for these analyses. Therefore, increasing the memory size of the annealing machine in the future will positively impact a wide range of applications in bioinformatics.

Availability and requirements

Project home page: https://github.com/ymatsumoto/qualign
Operating system(s): Any platforms
Programming language: Python
Other requirements: dwave-neal, pyqubo
License: MIT Licence
Any restrictions to use by non-academics: No restrictions

Key Points

- A new computing architecture, the annealing machine, has emerged and is available for practical use.
- Computations on the annealing machine require constructing and optimizing quadratic unconstrained binary optimization (QUBO) formulations for the specific application.
- A QUBO formulation for solving sequence alignment was constructed based on dynamic programming, and our implementation, named qualign, demonstrated accuracy comparable to conventional methods.
- This is one of the first bioinformatics applications for annealing machines despite its current memory size limitations.

Funding

This work was supported by JSPS KAKENHI Grant Number 19K16643.

Acknowledgements

We thank the Mitou Target project for giving us the development environment of annealing machines and its project managers, Dr Shu Tanaka, Dr Kotaro Tanahashi, Dr Yoshiki Matsuda, and Dr Ryo Tamura, for meaningful discussion and comments.

References

Aramon M, Rosenberg G, Valiante E, Miyazawa T, Tamura H, *et al.* (2019) Physics-Inspired Optimization for Quadratic Unconstrained Problems Using a Digital Annealer. Front. Phys. **7** http://dx.doi.org/10.3389/fphy.2019.00048

Boixo S, Rønnow TF, Isakov S V., Wang Z, Wecker D, *et al.* (2014) Evidence for quantum annealing with more than one hundred qubits. Nat. Phys. **10** (3), 218–224. http://dx.doi.org/10.1038/nphys2900

Crescenzi P, Goldman D, Papadimitriou C, Piccolboni A, Yannakakis M (1998) On the Complexity of Protein Folding. J. Comput. Biol. **5** (3), 423–465. http://dx.doi.org/10.1089/cmb.1998.5.423

Eddy SR (2004) Where did the BLOSUM62 alignment score matrix come from? Nat. Biotechnol. **22** (8), 1035–1036. http://dx.doi.org/10.1038/nbt0804-1035

Edgar RC (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res. **32** (5), 1792–1797. http://dx.doi.org/10.1093/nar/gkh340

Jünger M, Lobe E, Mutzel P, Reinelt G, Rendl F, *et al.* (2021) Quantum Annealing versus Digital Computing. ACM J. Exp. Algorithmics **26** (1), 1–30. http://dx.doi.org/10.1145/3459606

Lucas A (2014) Ising formulations of many NP problems. Front. Phys. 2 http://dx.doi.org/10.3389/fphy.2014.00005

Mardis ER (2011) A decade's perspective on DNA sequencing technology. Nature **470** (7333), 198–203. http://dx.doi.org/10.1038/nature09796

Medvedev P, Georgiou K, Myers G, and Brudno M (2007) Computability of Models for Sequence Assembly. In: Algorithms in Bioinformatics. Springer Berlin Heidelberg, Berlin, Heidelberg, Berlin, Heidelberg,pp. 289–301

Smith TF and Waterman MS (1981) Identification of common molecular subsequences. J. Mol. Biol. **147** (1), 195–197. http://dx.doi.org/10.1016/0022-2836(81)90087-5

Thompson J, Plewniak F, and Poch O (1999) BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs. Bioinformatics **15** (1), 87–88. http://dx.doi.org/10.1093/bioinformatics/15.1.87

Thompson JD, Higgins DG, and Gibson TJ (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res. **22** (22), 4673–4680. http://dx.doi.org/10.1093/nar/22.22.4673

Wang L and Jiang T (1994) On the Complexity of Multiple Sequence Alignment. J. Comput. Biol. **1** (4), 337–348. http://dx.doi.org/10.1089/cmb.1994.1.337

Yoshimura C, Hayashi M, Takemoto T, and Yamaoka M (2020) CMOS Annealing Machine: A Domain Specific Architecture for Combinatorial Optimization Problem. In: 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC).pp. 673–678