Subgroup Meeting #2

Hao-Yang Yen

NTHU, Interdisciplinary Program of Sciences

2024/7/15

Outline

1. Genetic algorithms (GAs), which simulate stochastic evolutionary dynamics, are powerful tools used across various interdisciplinary research fields.
2. The computational complexity of GAs increases significantly when the evolutionary dynamics become complex.
3. There are also critical behaviors in the genetic algorithms.
4. **Goal:** Apply the TN approach to GAs to reduce this computational complexity.

Genetic Algorithms

1. **Step 0: Generate Random Chromosomes**
   Initially, we create a set of chromosomes of equal length. The more chromosomes generated and the longer their lengths, the better the potential results.
2. **Step 1: Perform Crossover (Mating)**
   We randomly select two chromosomes and exchange several genes—specific numbers at corresponding positions—to create a new pair of chromosomes. This crossover can occur multiple times.

Genetic Algorithms

1. **Step 2: Apply Mutation**
   In this phase, we introduce genetic mutation. We randomly select some chromosomes and modify a few of their genes, resulting in new chromosomes. After this step, we will have double the number of chromosomes compared to the beginning.
2. **Step 3: Evaluate Fitness**
   Here, we define an evolutionary function that assigns a real number to each chromosome, indicating its fitness. We then select the top half of the chromosomes based on this evaluation, returning to the same number of chromosomes as in Step 1.

Genetic Algorithms

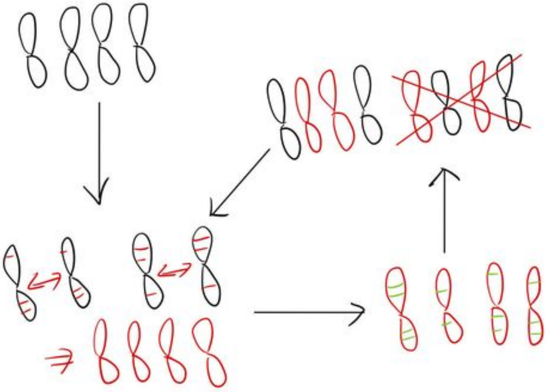

Figure: Genetic Algorithms.

Example

1. **Step 0: Generate Random Chromosomes**
   First generate several binary column vectors (chromosomes) of equal length N randomly.

$$|\phi_1\rangle, \cdots, |\phi_n\rangle \in \{0,1\}^N.$$

2. **Step 1: Perform Crossover (Mating)**
   Arrange all the chromosomes in pairs and choose several entries of them (genes) pairwise. Pairwise exchange these entries.

$$|\phi_i\rangle = \begin{pmatrix} \vdots \\ 0 \\ \vdots \end{pmatrix} \leftrightarrow \begin{pmatrix} \vdots \\ 1 \\ \vdots \end{pmatrix} = |\phi_j\rangle.$$

Example

1. **Step 2: Apply Mutation**
   Randomly select some chromosomes and change several entries of them.

$$|\phi_i\rangle = \begin{pmatrix} \vdots \\ 0 \\ \vdots \end{pmatrix} \to \begin{pmatrix} \vdots \\ 1 \\ \vdots \end{pmatrix}$$

2. **Step 3: Evaluate Fitness**
   Define a perfect chromosome $|\phi_0\rangle$, it is natural to define the evolutionary function

$$f : \{0,1\}^N \to \{0, \cdots, N\},\ |\phi_i\rangle \mapsto \|\,|\phi_i\rangle - |\phi_0\rangle\,\|_1.$$

   Select the top half of the chromosomes based on the evolutionary function.

Example

1. The animation of the simulation result.

The Critical Behaviors

1. We can characterize the algorithm by two parameters:

$$\alpha: \text{ The ratio of crossover,}$$

$$\beta: \text{ The ratio of mutation.}$$

2. It is therefore natural to consider it as a critical system. (reference: *Phase transitions and symmetry breaking in genetic algorithms with crossover*)

Tensor Network

1. **The Configuration Space**
   Since the sequence in this problem is binary, we can define the configuration space of this problem is

$$|\Phi(t)\rangle = \sum_{\{s_i\}} \Phi_t(s_1, \cdots, s_N) \, |s_1, \cdots, s_N\rangle \,,$$

   where $s_i \in \{0, 1\}$ and Φ_t is the evolutionary function that we want to minimized.

2. **The Hamiltonian**
   The mathematical structure of the GAs is still not very explicit. I still need to find some more references to construct it.

Tensor Network

1. **MPS Representation**
   Since Φ_t is a huge tensor, we employ SVD to decompose Φ_t into numerous pieces of little tensors. We have the MPS representation of the state
   $$|\Phi(t)\rangle = \sum_{\{s_i\}} \prod_i T^{[s_i]} |s_1, \cdots, s_N\rangle .$$
2. **Utilize DMRG Algorithm**
   Employ the DMRG algorithm to find the minimized evolutionary function, that is find
   $$\min_{|\Phi\rangle} \frac{\langle\Phi|\hat{H}|\Phi\rangle}{\langle\Phi|\Phi\rangle}$$
   and therefore the optimized state $|\Phi_0\rangle$.

References

1. *Efficient simulations of epidemic models with tensor networks: application to the one-dimensional SIS model*
2. *Phase transitions and symmetry breaking in genetic algorithms with crossover*