

Kuantum Hesaplama ile Amino Asit Dizilimlerinin Hamiltonian Analizi

Hamiltonian Analysis of Amino Acid Sequences by Quantum Computing

Sevdanur Genç 

Dr., Kastamonu Üniversitesi, Taşköprü Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, Kastamonu, Türkiye, e-mail: sgec@kastamonu.edu.tr

Öz

Moleküler biyolojide ve biyoinformatikteki hızlı ilerlemeler, genomik verilerin analizi ve işlenmesi alanında büyük bir ihtiyacı beraberinde getirmiştir. Bu ihtiyaç, geleneksel bilgisayarlarla işlenmekte olan genetik bilginin büyüklüğü ve karmaşıklığı göz önünde bulundurulduğunda, mevcut hesaplama kapasitesini zorlamaktadır. Bu makale, kuantum teknolojisinin genetik bilgi işlemeye nasıl katkı sağlayabileceğini incelemektedir. Kuantum bilgisayarlar, paralel işleme yetenekleri sayesinde büyük ve karmaşık genetik veri setlerini daha hızlı ve verimli bir şekilde işlemek için potansiyel sunmaktadır. Özellikle, amino asit dizilimlerinin analizi ve dönüşümü gibi biyoinformatik görevler, kuantum hesaplamaların potansiyel uygulama alanları arasındadır. Bu makale, temel seviyede amino asit dizilimlerinin nasıl kuantum devrelerine dönüştürülebileceğini ve kuantum teknolojisinin bu dönüşüm sürecinde Hamiltonian'ın rolünü ayrıntılı bir şekilde ele almaktadır. Ayrıca, makalede kuantum devrelerinin ve algoritmalarının kullanımıyla elde edilen sonuçlar ve örnekler bulunmaktadır. Bu çalışmanın amacı, kuantum teknolojisinin biyoinformatik ve moleküler biyoloji alanındaki potansiyelini vurgulamak ve şuan ki genetik bilgi işlemeyi nasıl dönüştürebileceğimize dair Türkçe literatüre bir perspektif sunmaktır.

Anahtar kelimeler: Kuantum Teknolojisi, Amino Asit Dizilimleri, Biyoinformatik, Kuantum Hesaplama, Kuantum Algoritma ve Kuantum Devreleri

Citation/Atf: GENÇ, S. (2023). Kuantum Hesaplama ile Amino Asit Dizilimlerinin Hamiltonian Analizi. *Kuantum Teknolojileri ve Enformatik Araştırmaları*. 1(1): 7-16, DOI: 10.70447/ktve.2234

Corresponding Author/ Sorumlu Yazar:

Sevdanur Genç

E-mail: sgec@kastamonu.edu.tr



Bu çalışma, Creative Commons Atif 4.0 Uluslararası Lisansı ile lisanslanmıştır.

This work is licensed under a Creative Commons Attribution 4.0 International License.

Abstract

Rapid advances in molecular biology and bioinformatics have created a great need for the analysis and processing of genomic data. This need challenges current computational capacity, given the size and complexity of genetic information being processed by conventional computers. This paper explores how quantum technology can contribute to genetic information processing. Quantum computers offer the potential to process large and complex genetic data sets faster and more efficiently thanks to their parallel processing capabilities. In particular, bioinformatics tasks such as the analysis and transformation of amino acid sequences are among the potential application areas of quantum computing. This article discusses in detail how amino acid sequences can be transformed into quantum circuits at the fundamental level and the role of quantum technology and the Hamiltonian in this transformation process. In addition, the paper includes results and examples obtained through the use of quantum circuits and algorithms. The aim of this paper is to highlight the potential of quantum technology in bioinformatics and molecular biology and to provide a perspective to the Turkish literature on how we can transform current genetic information processing.

Keywords: Quantum Technology, Amino Acid Sequences, Bioinformatics, Quantum Computing, Quantum Algorithms and Quantum Circuits

1. GİRİŞ

Günümüzde bilim ve teknoloji, hayatın her alanında önemli bir evrim yaşamaktadır. Bu evrim, özellikle biyoinformatik ve moleküler biyoloji alanlarında, kuantum bilgisayarların yükselişi ile daha da derinleşmektedir. Kuantum programlamanın sağladığı güçlü hesaplama yetenekleri, biyoloji dünyasının temel unsurlarından biri olan amino asit dizilimlerinin dönüşümü konusunda yeni kapılar açmaktadır. Bu makalede, kuantum programlamanın biyoinformatik ve moleküler biyoloji alanlarına getirdiği potansiyeli ve özellikle amino asit dizilimlerinin dönüşümü üzerindeki etkisini keşfedeceğiz. Kuantum bilgisayarların gelişimiyle, biyolojik sistemlerin daha iyi anlaşılması ve gelecekteki biyoteknoloji ve ilaç tasarımı projelerinin daha etkili bir şekilde gerçekleştirilmesi konusunda heyecan verici yeni olanaklar sunulmaktadır.

Moleküler biyoloji ve biyoinformatik alanlarında hızla artan veri miktarı, genetik bilginin analizi ve işlenmesinde büyük bir zorluk ve fırsat yaratmıştır. Özellikle, genetik bilginin temel taşı olan amino asit dizilimlerinin analizi, tıp, biyoteknoloji, ilaç geliştirme ve daha birçok alanda önemli sonuçlar doğurabilecek bir araştırma alanı olarak ortaya çıkmıştır. Ancak, genetik dizilerin karmaşıklığı ve büyüklüğü, geleneksel bilgisayarlarla bu verilerin hızlı ve verimli bir şekilde işlenmesini zorlaştırmaktadır.

Bu bağlamda, kuantum teknolojisi, genetik bilgi işlemenin geleceğine ışık tutabilecek bir potansiyele sahiptir. Kuantum bilgisayarlar, geleneksel bilgisayarlara kıyasla büyük ve karmaşık veri setlerini işleme konusunda önemli avantajlar sunabilirler. Bu avantajlar, özellikle amino asit dizilimlerinin analizi gibi biyoinformatik görevlerde büyük önem taşır. Kuantum bilgisayarlar, doğada kuantum mekaniği prensiplerine dayandıkları için geleneksel bilgisayarlarla mümkün olmayan paralel işlem yetenekleri sunarlar. Bu özelliği sayesinde, genetik verilerin hızlı ve hassas analizi mümkün hale gelir.

Bu makale, kuantum teknolojisinin moleküler biyoloji ve biyoinformatik alanlarına nasıl bir katkı sağlayabileceğini incelemek amacıyla yazılmıştır. Özellikle, amino asit dizilimlerinin kuantum devreleri ile nasıl dönüştürülebileceğini ve kuantum hesaplamaların bu süreçteki rolünü ayrıntılı bir şekilde ele almaktadır. Makale aynı zamanda kuantum devrelerinin ve algoritmalarının kullanımıyla elde edilen sonuçları ve potansiyel avantajları da örneklerle açıklamaktadır.

Bu çalışmanın amacı, kuantum teknolojisinin biyoinformatik ve moleküler biyoloji alanındaki potansiyelini vurgulamak ve gelecekte genetik bilgi işleme yöntemlerini nasıl dönüştürebileceğimize dair bir perspektif sunmaktır. Makalemizin ilerleyen bölümleri, kuantum teknolojisinin genetik bilgi işleme alanındaki potansiyelini daha ayrıntılı bir şekilde inceleyecektir.

2. İLGİLİ ÇALIřMALAR

Son zamanlarda, kuantum hesaplamanın moleküler dinamiklerin analizi, ilaç tasarımı, protein katlanma problemleri ve amino asit dizilimlerinin dönüşümü gibi konulardaki potansiyel etkileri tartışılmaktadır. Bu alandaki önceki çalışmaların nasıl bir temel oluşturduğu ve gelecekteki arařtırmalara nasıl ilham verdiği de söz konusudur. Kuantum programlamanın biyoinformatik ve biyolojik bilimlerle ilgili literatürdeki en güncel ve ilginç uygulamalarına birkaç örnek verilmiştir. Bunlar;

Iriyama ve Ohya 2009'da yapmış oldukları çalışma ile genom veya amino asit dizilerinin hizalanmasını NP problemlerinden biri olarak Kabul etmiş ve hesaplama karmaşıklığını $O(LN)$ olarak vermiştir [1]. Bununla birlikte, çoklu hizalamaları gerçekleştirebilmek adına kuantum algoritması ve kaos bilgi dinamiği aracılığıyla bir algoritma önermişlerdir. Fox ve arkadaşları 2021'de yayınlamış oldukları çalışmayla kodon optimizasyonu için kuantum hesaplama teknolojisinden yararlanmanın potansiyel etkisini arařtırmışlardır [2]. Bir Kuantum Tavlayıcı (QA), aynı amaç fonksiyonu ile programlanmış standart bir genetik algoritma (GA) ile karşılaştırılmıştır. QA'nın optimum çözümleri belirlemede rekabetçi olduğu bulunmuştur. Geçit tabanlı sistemlerin faydası da bir simülatör kullanılarak değerlendirilmiş ve mevcut nesil cihazların gerçekçi sorunları çözmek için hem kübit sayısı hem de bağlantı açısından donanım gereksinimlerinden yoksun olmasına rağmen, gelecek nesil cihazların oldukça verimli olabileceği sonucuna varılmıştır. Wong ve Chang 2022 yılındaki yapmış oldukları çalışmayla, iki boyutlu kare kafes üzerinde hidrofobik-hidrofilik modelde, klasik muadiline göre ikinci dereceden bir hızlanma ile N amino asit uzunluğundaki herhangi bir dizi için problemi çözmek için bir kuantum algoritması geliřtirmişlerdir. Bu hızlanma Grover'ın kuantum arama algoritması kullanılarak elde edilir. Algoritma, keyfi uzunluktaki amino asit dizileri için kullanılabilir. Algoritmayı Qiskit SDK kullanılarak IBM Quantum'un qasm simülatöründe başarıyla simüle etmişlerdir [3]. Bu iki arařtırmanın yapmış oldukları diđer bir çalışma ise, klasik muadillerine göre kuadratik hızlanma ile vücut merkezli kübik kafes üzerinde üç boyutlu

hidrofobik-hidrofilik modelde PSP için hızlı bir kuantum algoritması geliřtirmişlerdir. Önerilen algoritmanın uygulanabilirliğini kanıtlamak için 21 ve 25 kübit içeren IBM kuantum simülatöründe problemi başarıyla çözdük. Teorik olasılık tahminlerini hesaplayarak istenen konformasyonu bulmada deneysel olarak elde edilen yüksek başarı olasılığını doğrulamışlardır [4].

3. MATERYEL VE YÖNTEM

Bu bölüm, kuantum teknolojisi ile amino asit dizilimlerinin dönüşümünü gerçekleřtirmek için kullanılan materyal ve yöntemleri ayrıntılı bir şekilde açıklar. Bu adım, Türkçe literatür için yeni ve heyecan verici arařtırma alanının temelini oluşturur ve okuyuculara, yapılan deneylerin ve analizlerin nasıl yürütüldüğüne dair bir içgörü sunar.

3.1. Amino Asit Dizilimleri

Proteinler, hücrelerimizin yapı taşlarıdır ve birçok biyolojik sürecin düzenlenmesi ve gerçekleştirilmesi için gereklidir. Amino asit dizilimleri, bir proteinin yapısını ve işlevini belirler. Bu nedenle, bu dizilimlerin doğru bir şekilde analiz edilmesi, hücresel süreçlerin ve organizmanın sağlığının anlaşılması için temel öneme sahiptir. Amino asit dizilimleri, genetik bilginin proteinler aracılığıyla nasıl ifade edildiğini gösterir. DNA'daki genetik bilgi, mRNA aracılığıyla amino asit dizilimlerine çevrilir ve bu dizilimler, sonunda işlevsel proteinlerin üretimini başlatır. Bu süreç, merkezi bir öneme sahip olan merkezi bir konsepttir ve genetik mühendislik ve biyoteknoloji gibi birçok uygulama alanında kullanılır. Amino asit dizilimlerinin biyoinformatikteki analizi, bu önemli moleküler bileşenlerin işlevlerini ve etkileşimlerini anlamamıza yardımcı olur.

Bu çalışma için temel veri kaynağı, çeşitli organizmalardan elde edilen amino asit dizilimleridir. Genom verilerinin belirlenmesi, amino asit dizilimlerinin çıkarılması ve bu verilerin temizlenmesi, bu çalışmanın başlangıç aşamasını oluşturur. Ayrıca, bu dizilimlerin doğru bir şekilde anlamlandırılması ve genetik bilginin hassas bir şekilde temsil edilmesi için gerekli olan tüm veri ön işleme adımları "Kuantum Teknolojisi ile Amino Asit Dizilimlerinin Dönüşümü" başlığı altında ayrıntılı olarak açıklanmıştır.

3.2. Kauntum Devreleri ve Algoritmaları

Kuantum bilgisayarlar, büyük miktarda genomik verinin daha hızlı işlenmesine olanak tanır. Genom analizi, hastalık teşhisi, kişiselleştirilmiş tıp ve genetik araştırmalar gibi birçok alanda önemlidir. Kuantum bilgisayarlar, genetik verilerin analizini hızlandırabilir ve daha hassas sonuçlar elde etmemizi sağlayabilir. Proteinlerin doğru bir şekilde katlanması, biyolojik işlevlerini yerine getirebilmeleri için kritiktir. Kuantum bilgisayarlar, proteinlerin katlanma problemini daha hızlı ve etkili bir şekilde çözmek için kullanılabilir. Kuantum bilgisayarlar, kimyasal reaksiyonların ve moleküler etkileşimlerin simülasyonlarını daha hassas bir şekilde gerçekleştirmemize yardımcı olabilir. Biyoinformatikte sıkça karşılaşılan optimizasyon problemleri, kuantum bilgisayarlar tarafından daha hızlı ve etkili bir şekilde çözülebilir. Örneğin, gen dizilimlerinin analizi, genetik mühendislik projeleri ve biyoteknolojik süreçlerin iyileştirilmesi gibi alanlarda optimizasyon problemleri bulunur.

3.3. Veri Analizi

Protein Data Bank (PDB), proteinlerin üç boyutlu (3D) yapılarını içeren bir veritabanıdır [5]. PDB, bilim insanlarının proteinlerin yapısını anlamalarına, incelemelerine ve analiz etmelerine yardımcı olan uluslararası bir kaynaktır. Bu kaynaklar, X-ışını kristalografisi, nükleer manyetik rezonans (NMR) spektroskopisi ve elektron mikroskopisi gibi farklı yöntemlerle elde edilebilmektedir. PDB'ye erişmek oldukça kolaydır ve çoğu veri ücretsiz olarak sunulur. Bilim insanları, bu veritabanını çeşitli bilgisayar programları ve yazılımlar aracılığıyla kullanarak protein yapısı ile ilgili analizler yapabilirler. PDB, veri kalitesini yüksek tutmak için katı bir kalite kontrol sürecine sahiptir. Ayrıca, yeni protein yapıları keşfedildikçe ve çözümlendikçe sürekli güncellenir.

Dizilim, protein ya da DNA/RNA dizilimi şeklinde alınır. FASTA (Fast-All) formatı, biyoinformatik alanında genetik dizilerin ve protein dizilerinin temsil edilmesi için yaygın olarak kullanılan bir veri formatıdır.

FASTA format [6], genetik bilgiyi düzenlemek, depolamak ve paylaşmak için kullanılır ve birçok biyoinformatik yazılımı/aracı tarafından

desteklenir. FASTA formatı, bir dosya içinde birden fazla genetik veya protein dizisi girdisini destekler. Her girdi, başlık satırı ile başlar ve ardışık dizi satırları ile devam eder. Bu, birçok genetik veya protein dizisinin aynı dosya içinde düzenli bir şekilde saklanmasını sağlar. FASTA formatı, genetik dizilerin ve protein dizilerinin depolanması, paylaşılması, analiz edilmesi ve işlenmesi için standart bir araçtır. Bu format, BLAST (Basic Local Alignment Search Tool) gibi birçok biyoinformatik aracın giriş ve çıkış veri formatı olarak kullanılır ve genetik araştırmalarda yaygın olarak kullanılır.

Bu format, bir başlık (header) satırı ve bir dizi satırından oluşur ve temel bilgileri içerir. FASTA formatındaki her girdi, bir başlık satırı ile başlar. Başlık satırı, verinin neyi temsil ettiğini veya hangi organizmadan geldiğini açıklar. Başlık satırının genellikle bir '>' işareti ile başladığı ve hemen ardından bir tanım veya kimlik bilgisi geldiği görülür.

3.4. Kuantum Teknolojisi ile Amino Asit Dizilimlerinin Dönüşümü

Amino asit dizisini ikili diziler halinde kodluyor ve her biri için bir kuantum devresi oluşturuluyor. Ardından, her devre için bir Pauli operatörleri listesi oluşturulur, her kubit için bir tane ve onları birlikte çarparak bir Hamiltonian elde edilir. Sonrasında, her bir devre için ortaya çıkan kuantum durumları yazılır. Bu operatörleri ve Hamiltonianları oluşturmak için sırasıyla *qiskit.quantum_info* ve *qiskit.opflow*'daki *Pauli* ve *PauliOp* sınıfları kullanılır. Devreleri simüle etmek ve kuantum durumlarını hesaplamak için *qiskit*'in *Aer* modülü kullanılır. Tüm bu işlemler dört adımdan oluşmaktadır. Bunlar:

Adım 1: Amino Asit Diziliminin Hazırlanması

İlk adım, analiz edilmek istenen amino asit diziliminin hazırlanmasıdır. Bu çalışmada, örnek olarak Avrupa Biyoinformatik Enstitüsü tarafından oluşturulan bir PDB'dan "*Hemoglobin subunit alpha*" isimli bir dizilim FASTA formatında alınmıştır [7]. Hemoglobin subunit alpha, bir hemoglobin molekülünün yapı taşlarından biridir. Hemoglobin, kanımızdaki oksijen taşıma ve taşıma işlevini yerine getiren bir protein kompleksi olan tetramerik bir yapıya sahiptir. Bu yapıdaki her bir tetramer, iki alfa (alpha) alt birimi

ve iki beta (beta) alt birimini içerir. Hemoglobin subunit alpha, oksijen moleküllerini akciğerlerden dokulara taşıırken önemli bir rol oynar ve oksijen moleküllerini tutarak hemoglobinin işlevini gerçekleştirir. Her bir hemoglobin molekülü dört hemoglobin subunit alpha ve dört hemoglobin subunit beta içerir. Alınan FASTA formatında bulunan başlık satırının hemen ardından, genetik veya protein dizisi kendisi gelir. Dizi satırları, genellikle tek harfli amino asitlerin veya nükleotitlerin dizilerini içerir. Bu satırlar, başlık satırında belirtilen veriyi temsil eder. Hemoglobin protein dizisine ait FASTA dizisi Şekil 1'de verilmiştir.

Şekil 1. Hemoglobin protein dizisine ait FASTA sekansı

```
>pdb|1c7c|A
MLSPADKTNVKAANGKVGAGHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALT
NAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVST
VLTSTKYRGLVSPADKTNVKAANGKVGAGHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHG
KKVADALTNVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLD
KFLASVSTVLTSTKYR
```

Adım 2: Verilerin Kuantum Hesaplama İçin Hazırlanması

Amino asit dizilimi, kuantum hesaplama için uygun bir veri formatına dönüştürülmelidir. Kuantum bilgisayarların kullandığı kubitler (quantum bits) ile çalışıldığı için, veriyi bu kubitler ile temsil etmek gerekir.

Amino asitlerin bir DNA dizisine kodlanmasını simüle etmek için kullanılan basit bir algoritma tasarlanır. Verilen bir amino asit kodlama sözlüğü ve bir giriş DNA dizisi (Şekil 1'de bulunan hemoglobin verisi bir metin belgesi ile alınır) kullanılarak, DNA dizisindeki baz çiftlerini amino asitlere dönüştürür. Her amino asit iki bitlik bir ikili değere eşitlenir. Örneğin, Çizelge 1'de gösterildiği gibi 'A' amino asiti '00' olarak kodlanmıştır. Bu, DNA dizisindeki her iki bazın bir amino aside karşılık geldiği bir kodlama şemasıdır.

Çizelge 1. Aminoasitlerin bitlik kodlamaya çevirilmesi

| Amino Asit | Bitlik Eşitliği | Amino Asit | Bitlik Eşitliği |
|------------|-----------------|------------|-----------------|
| A | 00 | M | 10 |
| C | 01 | N | 11 |
| D | 10 | P | 00 |
| E | 11 | Q | 01 |
| F | 00 | R | 10 |
| G | 01 | S | 11 |
| H | 10 | T | 00 |
| I | 11 | V | 01 |
| K | 00 | W | 10 |
| L | 01 | Y | 11 |

DNA dizisini iki karakterlik çiftler halinde böler. Yani, 'AATCCGG' gibi bir dizi 'AA', 'TT', 'CC' ve 'GG' gibi ikili çiftlere ayrılır. Her iki baz çiftini amino asitlere dönüştürmek için bu ikili çiftleri ilgili amino asit kodlarına dönüştürür. Örneğin, 'AA' -> '00' + '00' şeklinde bir dönüşüm yapar. Bu kodlama yöntemi, DNA dizileri ve amino asitler arasındaki dönüşümü basit bir şekilde simüle eder ve çıktıda her baz çiftinin amino asit koduyla ilişkilendirilmiş olduğunu gösterir.

Adım 3: Kuantum Devrelerinin Tasarımı

Bu çalışmada, python programlama dilinde kuantum hesaplamaları için Qiskit [8] program kullanılmıştır. Bu program, belirli bir veri dizisi üzerinde kuantum devrelerini oluşturur. Bu çalışma için, devreler Hamiltonian [9] ile hesaplanır. Kuantum programlamada, Hamiltonian enerji seviyelerini ve sistemlerin dinamiklerini tanımlayan önemli bir kavramdır. Ancak, kuantum mekaniği bağlamında Hamiltonian, klasik mekanikten farklı bir rol oynar. Kuantum programlama da Hamiltonian, kuantum devrelerinin oluşturulması, kubitlerin enerji seviyeleri ve kubitler arası etkileşim gibi bir çok amaçla kullanılabilir. Aynı zamanda, kuantum bilgisayarlarla çeşitli hesaplamaların yapılmasına ve özellikle karmaşık kimya ve malzeme bilimi problemlerini çözmek için kullanılır. Hamiltonian'ın özdeğerleri ve özvektörleri, kuantum sistemlerin temel özelliklerini ve enerji seviyelerini incelemek için kullanılır. Hamiltonian, bir kuantum sistemdeki enerji seviyelerini ve etkileşimleri temsil eden bir matematiksel ifadedir.

Kuantum programlama içinde “Hamiltonian” veya “Hamilton operatörü”, kuantum sistemlerin enerjilerini ve evrimlerini tanımlayan önemli bir kavramdır. Hamiltonian, klasik fizikteki Hamilton fonksiyonuna benzer bir rol oynar, ancak kuantum sistemler için uyarlanmış bir operatördür. Kuantum mekaniğinde, sistemlerin dinamiklerini ve enerji seviyelerini açıklamak için kullanılır. Hamiltonian (H), bir kuantum sisteminin enerjilerini ve davranışlarını matematiksel olarak ifade eden bir operatördür. Genellikle $H = T + V$ şeklinde gösterilir. Burada: T, kinetik enerjiyi temsil eden kinetik terimi ifade eder. Bu terim, sistemin momentumu ve kinetik hareketini tanımlar. V, potansiyel enerjiyi temsil eden potansiyel terimi ifade eder. Bu terim, sistemin konumuna bağlı enerjiyi tanımlar. Hamiltonian operatörü, kuantum sistemlerin Schrödinger denklemiyle ilişkilendirilir. Eşitlik 1’de verilen Schrödinger denklemi [10], bir kuantum sisteminin zaman içindeki evrimini açıklamaktadır.

$$H|\psi\rangle = i\hbar \partial|\psi\rangle/\partial t \quad (1)$$

Burada:

- H : Hamiltonian operatörüdür.
- $|\psi\rangle$: sistem tarafından tanımlanan dalga fonksiyonunu ifade eder.
- I : karmaşık birimdir.
- \hbar : indirgenmiş Planck sabitidir.

Şekil 2. Qiskit kullanılarak kuantum devrelerinin tasarlanması

```
circuits = []
for binary in sequence:
    circuit = QuantumCircuit(len(binary), len(binary))
    for i, bit in enumerate(binary):
        if bit == '1':
            circuit.x(i)
    circuits.append(circuit)

paulis = []
for j in range(len(binary)):
    pauli_label = 'X'
    pauli = Pauli(pauli_label)
    pauli_op = PauliOp(pauli)
    paulis.append(pauli_op)

hamiltonian = paulis[0]
for j in range(1, len(paulis)):
    hamiltonian = hamiltonian @ paulis[j]

print(f"Circuit {i} Hamiltonian:")
print(hamiltonian)
```

Şekil 2’de amino asit dizilimi için oluşturulmuş sözlük yapılarının kuantum devrelerine dönüştürülmesi için geliştirilen qiskit algoritması yer almaktadır.

Çalışmada, oluşturulan ikili dizeleri kübitlere eşleyen sözlük yapıları oluşturulur. Bu sözlükte, iki bitlik ikili dizelerin kuantum operatörlerine nasıl dönüştürüleceği tanımlanır. Örneğin, ‘00’ -> 0, ‘01’ -> 1, ‘10’ -> 2 ve ‘11’ -> 3. Her ikili dize için bir kuantum devresi oluşturulur. Bu, her ikili dizideki ‘1’leri belirlemek için X kapısı kullanılarak yapılır. Örneğin, ‘1011’ ikili dizesi için 2. ve 4. pozisyonlardaki ‘1’leri belirtmek için X kapıları uygulanır. Oluşturulan kuantum devreleri ‘circuits’ isimli bir listede saklanır. Her ikili dize için bir dizi Pauli operatörü oluşturulur.

Pauli operatörleri [11] kuantum mekaniği ve kuantum bilgisayarlarında önemli bir rol oynayan matematiksel işlemcilerdir. Bu operatörler, temel kuantum kapılarını ve işlemlerini temsil etmek için kullanılır. Genellikle σ ile temsil edilirler ve dört ana Pauli operatörü vardır:

1. X Operatörü (σ_x): X operatörü, bir qubit’in x-eksenindeki spinini değiştirir. Eşitlik 2’de matris temsillemesi verilmiştir. Burada $|0\rangle$ ve $|1\rangle$, qubit’in temel durumlarını temsil etmektedir.

$$\sigma_x = |0\rangle\langle 1| + |1\rangle\langle 0| \quad (2)$$

2. Y Operatörü (σ_y): Y operatörü, bir qubit’in y-eksenindeki spinini değiştirir. Eşitlik 3’de matris temsillemesi verilmiştir. Burada “i”, karmaşık birimdir.

$$\sigma_y = -i(|0\rangle\langle 1|) + i(|1\rangle\langle 0|) \quad (3)$$

3. Z Operatörü (σ_z): Z operatörü, bir qubit’in z-eksenindeki spinini değiştirir. Eşitlik 4’de matris temsillemesi verilmiştir.

$$\sigma_z = |0\rangle\langle 0| - |1\rangle\langle 1| \quad (4)$$

4. Hadamard Operatörü (H Operatörü): Hadamard operatörü, bir qubit’i superpozisyon durumuna getirmek ve farklı durumlar arasında geçiş yapmak için kullanılır. Eşitlik 5’de matris temsillemesi verilmiştir.

$$H = \left(\frac{1}{\sqrt{2}}\right)(\sigma_x + \sigma_z) \quad (5)$$

Bu operatörler, her bir pozisyonda X veya I operatörlerini temsil eder. X operatörü, ‘1’ pozisyonundaki biti işlerken I operatörü ‘0’ pozis-

yonundaki biti temsil eder. Oluřturulan Pauli operatörleri, bir Hamiltonian'ı hesaplamak için birbiriyle çarpılır. Her bir ikili dize için hesaplanan Hamiltonian, ekrana yazdırılır. Bu, her ikili dizinin kuantum operatörleri tarafından temsil edilen enerji seviyelerini ve etkileşimlerini gösterir.

Adım 4: Kuantum İşleme

Kuantum bilgisayarlar, sürekli ve karışık bir şekilde 0 ve 1 değerleri arasında bulunan kuantum durumlarını kullanarak veriyi işlerler. Bu süreç, belirli bir kuantum devresi tarafından tanımlanan işlemler aracılığıyla gerçekleştirilir. Oluřturulan listede kuantum devreleri tanımlanmıştır, bu devreler bir simülatör üzerinde döngü aracılığıyla tek tek çalıştırılır. Şekil 3'de bu döngüye ait algoritma yer almaktadır.

Şekil 3. Her devre için ortaya çıkan kuantum durumlarını listeleyen algoritma

```
for i, circuit in enumerate(circuits):
    print(f"Circuit {i}:")
    print(circuit)
    backend = Aer.get_backend('statevector_simulator')
    result = execute(circuit, backend).result()
    print(result.get_statevector())
```

`backend = Aer.get_backend('statevector_simulator')` kodu, Aer adlı Qiskit modülünü kullanarak bir kuantum simülatörünü belirler. `'statevector_simulator'` parametresi, kuantum devrelerinin durum vektörlerini hesaplamak için kullanılan bir simülatör türünü temsil eder. Bu, her bir devrenin sonucunun hesaplanması için kullanılacak arka uç (backend) cihazını tanımlar. `result = execute(circuit, backend).result()` kodu, belirtilen kuantum devreyi ve simülatörü kullanarak hesaplamayı başlatır ve sonucu `result` adlı bir değişkende saklar. `result.get_statevector()` kodu, kuantum devrenin sonucu olan durum vektörünü ekrana yansıtmak için kullanılır. Bu, her bir devrenin sonucunda bulunan kuantum durumunu gösterir. Kuantum durum, tüm kübitlerin olası durumlarını içerir ve kuantum devresinin sonucunu temsil eder.

4. DENEYSEL SONUÇLAR

Bu bölüm, kuantum teknolojisinin amino asit dizilimlerinin dönüşümü üzerindeki etkisini değerlendirmek için verilen deneysel çalışmanın sonuçlarını sunmaktadır. Hazırlanan algoritmayla birlikte yapılan çalışmada, protein veri

bankasından elde edilen Hemoglobin protein dizisinin toplamda 283 amino asit bulunmaktadır. Şekil 1'de ilgili protein dizisinin FASTA formatı verilmişti. Qiskit çerçevesi ile birlikte Hemoglobin dizisi öncelikle FASTA formatından bitlik formata dönüřtürülmüřtür. Şekil 1'de verilen Hemoglobin dizisinin bu kurallar doğrultusunda bitlik dönüşümü Çizelge 2'de verilmiştir.

Çizelge 2'de her bir amino asit sırayla çiftli olacak şekilde bitlik formatına dönüřtürülmüřtür. Toplamda 141 adet çift oluşup sonuncu amino asit olan R çifti olacak bir amino asit olmadığı için değerlendirilmeye alınmamıştır. Deney sonucunda toplamda 141 adet kuantum devresi ve beraberinde durum vektör (statevector) listesi oluşturulmuřtur. Böylelikle, kuantum işlemi tamamlanmış ve sonuçlar elde edilmiştir. Ayrıca, her bir devrenin sonucunda elde edilen kuantum durumlarına örnek olabilecek rastgele dört adet kuantum devresinin çıktısı şekil 4'de verilmiştir.

Aynı zamanda Şekil 4 (a)'da , 41 numaralı kuantum devresine ait devre çizimi ve durum vektör (statevector) listesinin çıktısı bir arada bulunmaktadır. Şekil 4 (b)'de 82, şekil 4 (c)'de 84 ve şekil 4 (d)'de de 114 numaralı kuantum devrelere ait devre çizimleri ve durum vektörleri örnek olarak verilmiştir. Geri kalan 137 kuantum devresi yine bu dört devreye benzer şekillerde üretilmiştir. Bu ve bunun gibi geliştirilmiş algoritmalar neticesinde alınan sonuçlar, kuantum teknolojisinin biyoinformatik alanında nasıl bir devrim yaratabileceği konusunda önemli ipuçları sunmaktadır.

Çizelge 2. Hemogloblin dizisinin bitlik dönüşümü

| | | | | | | |
|------------|------------|------------|------------|------------|------------|------------|
| ML -> 1001 | TY -> 0011 | SA -> 1100 | VH -> 0110 | AH -> 0010 | GK -> 0100 | KL -> 0001 |
| SP -> 1100 | FP -> 0000 | LS -> 0111 | AS -> 0011 | AG -> 0001 | KV -> 0001 | LS -> 0111 |
| AD -> 0010 | HF -> 1000 | DL -> 1001 | LD -> 0110 | EY -> 1111 | AD -> 0010 | HC -> 1001 |
| KT -> 0000 | DL -> 1001 | HA -> 1000 | KF -> 0000 | GA -> 0100 | AL -> 0001 | LL -> 0101 |
| NV -> 1101 | SH -> 1110 | HK -> 1000 | LA -> 0100 | EA -> 1100 | TN -> 0011 | VT -> 0100 |
| KA -> 0000 | GS -> 0111 | LR -> 0110 | SV -> 1101 | LE -> 0111 | AV -> 0001 | LA -> 0100 |
| AW -> 0010 | AQ -> 0001 | VD -> 0110 | ST -> 1100 | RM -> 1010 | AH -> 0010 | AH -> 0010 |
| GK -> 0100 | VK -> 0100 | PV -> 0001 | VL -> 0101 | FL -> 0001 | VD -> 0110 | LP -> 0100 |
| VG -> 0101 | GH -> 0110 | NF -> 1100 | TS -> 0011 | SF -> 1100 | DM -> 1010 | AE -> 0011 |
| AH -> 0010 | GK -> 0100 | KL -> 0001 | KY -> 0011 | PT -> 0000 | PN -> 0011 | FT -> 0000 |
| AG -> 0001 | KV -> 0001 | LS -> 0111 | RG -> 1001 | TK -> 0000 | AL -> 0001 | PA -> 0000 |
| EY -> 1111 | AD -> 0010 | HC -> 1001 | VL -> 0101 | TY -> 0011 | SA -> 1100 | VH -> 0110 |
| GA -> 0100 | AL -> 0001 | LL -> 0101 | SP -> 1100 | FP -> 0000 | LS -> 0111 | AS -> 0011 |
| EA -> 1100 | TN -> 0011 | VT -> 0100 | AD -> 0010 | HF -> 1000 | DL -> 1001 | LD -> 0110 |
| LE -> 0111 | AV -> 0001 | LA -> 0100 | KT -> 0000 | DL -> 1001 | HA -> 1000 | KF -> 0000 |
| RM -> 1010 | AH -> 0010 | AH -> 0010 | NV -> 1101 | SH -> 1110 | HK -> 1000 | LA -> 0100 |
| FL -> 0001 | VD -> 0110 | LP -> 0100 | KA -> 0000 | GS -> 0111 | LR -> 0110 | SV -> 1101 |
| SF -> 1100 | DM -> 1010 | AE -> 0011 | AW -> 0010 | AQ -> 0001 | VD -> 0110 | ST -> 1100 |
| PT -> 0000 | PN -> 0011 | FT -> 0000 | GK -> 0100 | VK -> 0100 | PV -> 0001 | VL -> 0101 |
| TK -> 0000 | AL -> 0001 | PA -> 0000 | VG -> 0101 | GH -> 0110 | NF -> 1100 | TS -> 0011 |
| | | | | | | KY -> 0011 |

Şekil 4. Kuantum devrelerine ait devre çizimi ve durum vektör (statevector) listeleri

```
Circuit 41:
q_0: _____
q_1:  [ X ]
q_2:  [ X ]
q_3:  [ X ]
c: 4/_____

Statevector([0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
1.+0.j, 0.+0.j],
dims=(2, 2, 2, 2))
```

(a)

```
Circuit 82:
q_0:  [ X ]
q_1:  [ X ]
q_2:  [ X ]
q_3:  [ X ]
c: 4/_____

Statevector([0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 1.+0.j],
dims=(2, 2, 2, 2))
```

(b)

```
Circuit 84:
q_0:  [ X ]
q_1:  [ X ]
q_2: _____
q_3: _____
c: 4/_____

Statevector([0.+0.j, 0.+0.j, 0.+0.j, 1.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 0.+0.j],
dims=(2, 2, 2, 2))
```

(c)

```
Circuit 114:
q_0:  [ X ]
q_1: _____
q_2: _____
q_3: _____
c: 4/_____

Statevector([0.+0.j, 1.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j,
0.+0.j, 0.+0.j],
dims=(2, 2, 2, 2))
```

(d)

5. SONUÇ VE TARTIŐMA

Bu alıřmanın temel amacı, kuantum teknolojisinin amino asit dizilimlerinin dnüşümünde nasıl kullanılabileceğini göstermektir. Amino asit dizilimleri, FASTA formatından iki bitlik formata dnüşürüldükten sonra kuantum devreleri oluşturulmuştur. Kullanılan Qiskit çerçevesi ile birlikte python programlama dilinde ilgili kütüphaneler kullanılmıştır. Özellikle Hamiltonian hesaplamak için Pauli operatörleri kullanılarak bir algoritma oluşturulmuştur. Makalede, kuantum programlamanın biyoinformatik ve moleküler biyoloji alanlarına getirdiđi potansiyeli inceledik ve özellikle amino asit dizilimlerinin dnüşümü üzerindeki etkisini arařtırdık. Kuantum programlamanın, bu alanlarda hızlı ve etkili çzmler sunabileceđi görüldü.

Kuantum hesaplama, geleneksel bilgisayarlarla karşılaştırıldığında, çok büyük moleküler sistemlerin daha hızlı ve hassas bir şekilde simüle edilmesini sağlar. Bu, özellikle ilaç tasarımı ve protein katlanma problemleri gibi karmaşık sorunların çözümünde büyük bir potansiyel sunar. Ayrıca, kuantum programlama, amino asit dizilimlerinin hızlı bir şekilde karşılaştırılması ve analiz edilmesi için yeni yöntemler geliřtirmemizi sağlar. Kuantum bilgi işlem, temel olarak özellikle büyük veri setleri ve kompleks optimizasyon problemleri gibi alanlarda büyük bir atılım yapabilir. Ancak, kuantum bilgisayarlar henüz erken aşamada olduklarından, pratik uygulamaların tam potansiyeline ulaşabilmesi için daha fazla geliřtirme ve optimize edilmesi gerekmektedir. Ayrıca, bu makalede ele alınan konuların yanı sıra, kuantum programlamanın biyoinformatik, tıp ve biyokimya gibi alanlarda daha geniş bir etki yaratabileceđi birçok başka uygulama alanı da vardır. Kuantum bilgisayarların geliřtirilmesi ve kullanılabilirliğinin artması, bu potansiyelin daha fazla keşfedilmesine ve uygulamalara dnüşürülmesine olanak tanır.

Sonuç olarak, kuantum programlama, biyoinformatik ve moleküler biyoloji alanlarına önemli katkılar sağlayabilir. Bu yeni teknolojinin, büyük ölçekli veri analizi, simülasyon ve optimizasyon problemlerinin çözümünde önemli bir rol oynayacağını ve gelecekte daha fazla inovasyonun kapılarını açacağı düşünölmektedir. Bu alandaki

arařtırmaların devamı, kuantum programlamanın bu potansiyelinin tam olarak anlaşılmasına katkı sağlayacaktır.

KAYNAKÇA

- [1] S. Iriyama ve M. Ohya, "On Quantum Algorithm for Multiple Alignment of Amino Acid Sequences", *Quantum Bio-Informatics II: From Quantum Information to Bio-Informatics*, ss. 92-104, Oca. 2009, doi: 10.1142/9789814273756_0009.
- [2] D. M. Fox, K. M. Branson, ve R. C. Walker, "mRNA codon optimization with quantum computers", *PLoS One*, c. 16, sy 10, s. e0259101, Eki. 2021, doi: 10.1371/JOURNAL.PONE.0259101.
- [3] R. Wong ve W. L. Chang, "Fast quantum algorithm for protein structure prediction in hydrophobic-hydrophilic model", *J Parallel Distrib Comput*, c. 164, ss. 178-190, Haz. 2022, doi: 10.1016/J.JPDC.2022.03.011.
- [4] R. Wong ve W. L. Chang, "Quantum speedup for protein structure prediction", *IEEE Trans Nanobioscience*, c. 20, sy 3, ss. 323-330, Tem. 2021, doi: 10.1109/TNB.2021.3065051.
- [5] S. K. Burley, H. M. Berman, G. J. Kleywegt, J. L. Markley, H. Nakamura, ve S. Velankar, "Protein Data Bank (PDB): The single global macromolecular structure archive", *Methods in Molecular Biology*, c. 1607, ss. 627-641, 2017, doi: 10.1007/978-1-4939-7000-1_26/COVER.
- [6] W. R. Pearson, "Using the FASTA program to search protein and DNA sequence databases.", *Methods Mol Biol*, c. 24, ss. 307-331, 1994, doi: 10.1385/0-89603-246-9:307/COVER.
- [7] "Hemoglobin subunit alpha in PDB entry 1c7c < PDBe < EMBL-EBI". Eriřim: 14 Ekim 2023. [Çevrimiçi]. Eriřim adresi: <https://www.ebi.ac.uk/pdbe/entry/pdb/1c7c/protein/1>
- [8] A. Cross, Cross, ve Andrew, "The IBM Q experience and QISKit open-source quantum computing software", *APS*, c. 2018, s. L58.003, 2018, Eriřim: 14 Ekim 2023. [Çevrimiçi]. Eriřim adresi: <https://ui.adsabs.harvard.edu/abs/2018APS..MARL58003C/abstract>
- [9] A. Ciani, B. M. Terhal, ve D. P. Divincenzo, "Hamiltonian quantum computing with supercon-

- ducting qubits”, *Quantum Sci Technol*, c. 4, sy 3, s. 035002, May. 2019, doi: 10.1088/2058-9565/AB18DD.
- [10] Rabinowitz ve P. H., “On a class of nonlinear Schrödinger equations”, *ZAMP*, c. 43, sy 2, ss. 270-291, Mar. 1992, doi: 10.1007/BF00946631.
- [11] P. Nema ve M. J. Nene, “Pauli Matrix based Quantum Communication Protocol”, *Proceedings of IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation, ICATMRI 2020*, Ara. 2020, doi: 10.1109/ICATMRI51801.2020.9398393.