

**ANKIRI KARATEKİN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**GROMACS TABANLI MD SİMÜLASYON HAZIRLIĞINA YÖNELİK
ARAYÜZ GELİŞTİRİLMESİ**

Hüseyin ŞAHİNER

ELEKTRONİK VE BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**ANKIRI
2025**

Her hakkı saklıdır

TEZ ONAYI

Hüseyin ŞAHİNER tarafından hazırlanan “**GROMACS Tabanlı MD Simülasyon Hazırlığına Yönelik Arayüz Geliştirilmesi**” adlı tez çalışması 10/03/2025 tarihinde aşağıdaki jüri tarafından oy birliği ile Çankırı Karatekin Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Mühendisliği Anabilim Dalında **Yüksek Lisans Tezi** olarak kabul edilmiştir.

Danışman : Doç. Dr. Mustafa KARHAN

Eş Danışman : Prof. Dr. Volkan EYÜPOĞLU

Jüri Üyeleri :

Başkan : Doç. Dr. Mustafa KARHAN
Elektronik ve Bilgisayar Mühendisliği Anabilim Dalı
Çankırı Karatekin Üniversitesi

Üye : Prof. Dr. Şevki ADEM
Kimya Anabilim Dalı
Çankırı Karatekin Üniversitesi

Üye : Dr. Öğr. Üyesi Fuat TÜRK
Bilgisayar Mühendisliği Anabilim Dalı
Kırıkkale Üniversitesi

Yukarıdaki sonucu onaylarım

Prof. Dr. Ersoy YILMAZ

Enstitü Müdürü

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Çankırı Karatekin Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğine göre hazırlamış olduğum “**GROMACS Tabanlı MD Simülasyon Hazırlığına Yönelik Arayüz Geliştirilmesi**” konulu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, tezin Çankırı Karatekin Üniversitesi Fen Bilimleri Enstitüsü’nden başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve bu çalışmanın Çankırı Karatekin Üniversitesi tarafından kullanılan “Bilimsel İntihal Tespit Programı”yla tarandığını, “intihal içermediğini” beyan ederim. Çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması halinde ortaya çıkacak tüm ahlaki ve hukuki sonuçlara razı olduğumu bildiririm. Çankırı Karatekin Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca gereğinin yapılmasını arz ederim (10/03/2025).

Hüseyin ŞAHİNER

ÖZET

Yüksek Lisans Tezi

GROMACS TABANLI MD SİMÜLASYON HAZIRLIĞINA YÖNELİK ARAYÜZ GELİŞTİRİLMESİ

Hüseyin ŞAHİNER

Çankırı Karatekin Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik ve Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Mustafa KARHAN
Eş Danışman: Prof. Dr. Volkan EYÜPOĞLU

Moleküler dinamik (MD) simülasyon, biyolojik ve kimyasal süreçlerin atomik düzeyde anlaşılmasında kritik bir yöntemdir. Ancak bu simülasyon, özellikle GROMACS gibi yazılımlar kullanılarak yapıldığında hem yazılımsal bilgi ve beceri hem de yüksek hesaplama gücü içeren donanım gerektirir. Bu durum, birçok araştırmacı için erişim engeli oluşturmakta ve bilimsel süreçleri yavaşlatmaktadır. Bu çalışmada, GROMACS tabanlı simülasyon süreçlerini kullanıcı dostu bir web ve bulut tabanlı platform ile otomatize ederek, simülasyon erişimini kolaylaştırmayı ve kullanıcı dostu bir arayüzün üretilmesi sağlanmıştır. Sistem ile kullanıcıların protein ve ligand dosyalarını yükleyerek simülasyonu başlatmaları, sonuçları görselleştirmeleri ve elde edilen sonuçların paylaşımı mümkün kılınmıştır. Böylece hem yazılımsal anlamda uzmanlığı sınırlı kullanıcıların bu alana girmeleri teşvik edilmiş hem de deneyimli araştırmacıların verimliliği artırılmıştır. Proje, web arayüzü, lokal GROMACS sunucuları ve bulut tabanlı bir altyapıyı bir araya getiren bir yöntemle yürütülmüştür. Web arayüzüne yüklenen dosyalar, GROMACS sunucularına iletilmiş ve internet üzerinden gerekli ek dosyalar indirilerek standart GROMACS iş akışı uygulanmıştır. Simülasyonun tamamlanmasının ardından, sistem otomatik olarak RMSD, enerji grafikleri ve atomik mesafe değişimleri gibi grafiksel çıktılar üretilmiştir. Bu çıktılar, bir bulut sunucusuna yüklenerek kullanıcıların sonuçlara kolayca erişimi hem web üzerinden hem de e-posta yoluyla sağlanmıştır.

2025, 64 sayfa

ANAHTAR KELİMELER: GROMACS, Moleküler dinamik simülasyon, Web tabanlı platform, Bulut entegrasyonu, Protein-Ligand etkileşimleri

ABSTRACT

Master of Science Thesis

DEVELOPMENT OF INTERFACE FOR GROMACS BASED MD SIMULATION PREPARATION

Hüseyin ŞAHİNER

Çankırı Karatekin University
Graduate School of Natural and Applied Sciences
Department of Electronics and Computer Engineering

Advisor: Assoc. Prof. Dr. Mustafa KARHAN

Co-Advisor: Prof. Dr. Volkan EYÜPOĞLU

Molecular Molecular dynamics (MD) simulation is a critical method for understanding biological and chemical processes at the atomic level. However, this simulation requires both software knowledge and skills and hardware with high computational power, especially when performed using software such as GROMACS. This creates an access barrier for many researchers and slows down scientific processes. In this study, we automated GROMACS-based simulation processes with a user-friendly web and cloud-based platform to facilitate simulation access and to produce a user-friendly interface. The system allows users to start the simulation by uploading protein and ligand files, visualize the results and share the results obtained. Thus, both users with limited software expertise were encouraged to enter this field and the productivity of experienced researchers was increased. The project was carried out using a combination of a web interface, local GROMACS servers and a cloud-based infrastructure. The files uploaded to the web interface were transmitted to the GROMACS servers and the standard GROMACS workflow was applied by downloading the necessary additional files over the internet. Upon completion of the simulation, the system automatically generated graphical outputs such as RMSD, energy plots and atomic distance changes. These outputs were uploaded to a cloud server, allowing users to easily access the results both on the web and via email.

2025, 64 pages

Keywords: GROMACS, Molecular dynamics simulations, Web-Based platform, Cloud integration, Protein-Ligand interactions

ÖNSÖZ VE TEŞEKKÜR

Bu tezin hazırlanmasında desteklerini esirgemeyen birçok kişi ve kuruma teşekkür etmeyi bir borç bilirim.

Öncelikle, tez sürecim boyunca bilgi ve tecrübelerini benimle paylaşarak beni yönlendiren, değerli katkılarıyla çalışmamın her aşamasında yanımda olan saygıdeğer danışman hocam Mustafa KARHAN'a şükranlarımı sunarım.

Araştırmalarım sırasında bana rehberlik eden ve önemli fikirler sunan Volkan EYÜPOĞLU'na içten teşekkür ederim.

Ayrıca, çalışmam boyunca maddi ve manevi destekleriyle yanımda olan sevgili ailem, eşim Aynur ve kızım Aysıla'ya sabır ve anlayışları için minnettarım.

Son olarak, bu süreçte beni motive eden ve desteğini hissettiren dostlarım ve çalışma arkadaşlarıma teşekkür ederim.

Tez çalışmamda emeği geçen herkese teşekkürlerimi ve saygılarımı sunarım.

Hüseyin ŞAHİNER

Çankırı, Şubat 2025

İÇİNDEKİLER

ÖZET.....	i
ABSTRACT	ii
ÖNSÖZ VE TEŞEKKÜR.....	iii
İÇİNDEKİLER	iv
SİMGELER DİZİNİ	vi
KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	ix
ÇİZELGELER DİZİNİ	x
1. GİRİŞ.....	1
2. LİTERATÜR ÖZETİ.....	4
2.1 Moleküler Dinamik Simülasyon Temelleri.....	4
2.2 GROMACS ve Diğer MD Yazılımlarının Karşılaştırması	5
2.2.1 GROMACS	7
2.2.2 NAMD	9
2.2.3 Maestro-Schrödinger.....	9
2.2.4 AMBER	10
2.2.5 OpenMM	10
2.2.6 CHARMM.....	10
2.3 Simülasyon Otomasyonu: Mevcut Yaklaşımlar	11
2.4 Literatürde Yapılan Çalışmalar	12
2.5 Python ve Shell Script Kullanımı Hakkında Temel Bilgiler	17
2.6 PHP-Laravel ve MySQL kullanımı Hakkında Temel Bilgiler.....	17
2.7 Visual Molecular Dynamics (VMD) Programı.....	18
2.8 UCSF Chimera Programı	19
3. MATERYAL VE METOT	21
3.1 Yazılımın Genel Mimarisi	22
3.2 Python ve Shell Script Entegrasyonu	24
3.3 GROMACS Komutlarının Otomasyonu	24
3.4 Geliştirme Süreci.....	25
3.5 Kodlama ve Test Aşamaları.....	26
3.6 Kullanıcı Arayüzü ve İşlevsellik	26

3.7 Yazılımın Kullanıcı Rehberi	31
3.8 Gromacs Simülasyon Protokolü	31
3.8.1 Topoloji oluşturulması	33
3.8.2 Simülasyon kutusu boyut tanımlama ve çözücü ekleme işlemleri	33
3.8.3 İyonları ekleme işlemleri.....	34
3.8.4 Enerji minimizasyonu	34
3.8.5 Sabit basınç ve hacim denge simülasyonu (nvt.run/npt.run)	35
3.8.6 Moleküler simülasyonun yürütülmesi (md.run).....	36
3.8.7 Analiz ve görselleştirme	36
4. BULGULAR VE TARTIŞMA	39
4.1 Örnek Çalışma	40
4.1.1 C-alpha RMSD	42
4.1.2 C-alpha RMSF	44
4.1.3 Bağlanma enerjisi	45
4.1.4 Frame vs MMGBSA	47
4.1.5 Ligand enerjisi	48
4.1.6 Ligand RMSD	50
4.1.7 Reseptör enerjisi	51
4.1.8 Radius of gyration	53
4.1.9 Solvent accessible surface area (SASA)	54
4.1.10 Ortalama ve toplam hidrojen bağı doluluk analizi.....	56
4.1.11 Sonuç analizi.....	58
4.2 Yazılımın Performans Testleri.....	60
4.3 İşlem Süresi ve Verimlilik.....	60
4.4 Hata Oranı ve Düzeltmeler	61
4.5 Belirli Molekül Sistemlerinde Yazılımın Uygulaması.....	62
5. ÖNERİLER.....	63
KAYNAKLAR	66
ÖZGEÇMİŞ.....	Hata! Yer işareti tanımlanmamış.

SİMGELER DİZİNİ

ΔG	Bağlanma afinitesi
C- α	C-alpha



KISALTMALAR DİZİNİ

API	Application Programming Interface
AWS	Amazon Web Services
CHARMM	Chemistry at HARvard Macromolecular Mechanics
CPU	Central Processing Unit
EM	Energy minimization
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HPC	High Performance Computing
kJ/mol	Molar Energy (Kilojoule / mol)
MMGBSA	Molecular Mechanics Generalized Born Surface Area
MOF	Metal-Organic Framework
MPI	Message Passing Interface
MVC	Model-View-Controller
nm	Nanometer
NMR	Nuclear Magnetic Resonance
NPT	Constant Number of Particles, Pressure, and Temperature
ns	Nanosecond
NVT	Constant Number of Particles, Volume, and Temperature
OPLS	Optimized Potentials for Liquid Simulations
ORM	Object-Relational Mapping
PDB	Protein Data Bank
PHP	Hypertext Preprocessor
ps	Picosecond
RAID	Redundant Array of Independent Disks
RMSD	Root Mean Square Deviation
RMSF	Root Mean Square Fluctuation
SASA	Solvent Accessible Surface Area
SDK	Software Development Kit
UCSF	University of California San Francisco
VMD	Visual Molecular Dynamics



ŞEKİLLER DİZİNİ

Şekil 2.1 NVIDIA firmasının developer NRF karşılaştırma grafiği (https://developer.nvidia.com , 2025).....	5
Şekil 2.2 NVIDIA GPU ve CUDA Core karşılaştırmaları (https://developer.nvidia.com , 2025).....	7
Şekil 2.3 GROMACS logosu (https://gromacs.org , 2025)	8
Şekil 2.4 NAMD 2.14 ile 3.0 versiyonu GPU performans karşılaştırması (https://www.ks.uiuc.edu , 2025).....	9
Şekil 2.5 MemBuilder arayüzü (Ghahremanpour et al,2014).	14
Şekil 2.6 CHAPHERONg çalışma mantığı (Yekeen et al,2023).	16
Şekil 2.7 (A) Python ile (B) ise UCSF Chimera ile oluşturulmuş Protein-Ligand örnek görseli	20
Şekil 3.1 Çalışmanın gelişme süreci	21
Şekil 3.2 Çalışmanın akış şeması	23
Şekil 3.3 Web sayfası görünümü - Anasayfa	27
Şekil 3.4 Web sayfası Giriş (Login) ve Kayıt (Register) görüntüleri	28
Şekil 3.5 Anasayfa (Dashboard)	29
Şekil 3.6 Proje oluşturma formu	30
Şekil 3.7 Proje detay ve sonuç izleme ekranı.....	31
Şekil 3.8 GROMACS işlem adımları (https://mdtutorials.com ,2025).....	32
Şekil 3.9 GROMACS akış şeması (https://gromacs.org ,2025).....	32
Şekil 3.10 Grafik verilerini oluşturmak için kullanılan Python kodu	38
Şekil 4.1 Örnek ligand görüntüleri.....	40
Şekil 4.2 Sistemlerin frame1.pdb dosyalarının ortak görselleri	41
Şekil 4.3 Çalışmaların ortak C-alpha RMSD grafiği	43
Şekil 4.4 Çalışmaların ortak C-alpha RMSF grafiği.....	44
Şekil 4.5 Çalışmaların ortak bağlanma enerjisi grafiği.....	46
Şekil 4.6 Çalışmaların ortak Frame ile MMGBSA karşılaştırma grafiği.....	47
Şekil 4.7 Çalışmaların ortak ligand enerjisi grafiği	49
Şekil 4.8 Çalışmaların ortak ligand RMSD grafiği.....	50
Şekil 4.9 Çalışmaların ortak reseptör enerjisi grafiği.....	52
Şekil 4.10 Çalışmaların ortak Rg grafiği	53
Şekil 4.11 Çalışmaların ortak SASA grafiği.....	55
Şekil 4.12 Çalışmaların ortalama ve toplam doluluk oranlarının grafikleri	57

ÇİZELGELER DİZİNİ

Çizelge 2.1 MD yazılımlarının bazı özellikleri bakımından karşılaştırılması	6
Çizelge 4.1 Çalışmaların ortalama ve toplam doluluk oranlarının çıkarımları	58
Çizelge 4.2 Çalışmaların protein-ligand komplekslerinin kapsamlı karakterizasyon tablosu	59



1. GİRİŞ

Moleküler dinamik (MD) simülasyonu, hesaplamalı kimya, biyofizik ve malzeme bilimi gibi birçok disiplinde yaygın olarak kullanılan temel bir tekniktir. Bu yöntem, moleküllerin atomik seviyedeki hareketlerini ve zaman içindeki davranışlarını modellemek için fiziksel ve matematiksel prensiplere dayanır. MD simülasyonları, moleküllerin belirli bir süre boyunca nasıl hareket ettiğini, birbirleriyle nasıl etkileşime girdiğini ve bu etkileşimlerin sistemin genel özelliklerini nasıl etkilediğini anlamamıza olanak tanır. Bu sayede, biyolojik süreçlerin moleküler mekanizmaları, ilaçların hedef proteinlerle olan etkileşimleri ve malzemelerin fiziksel ve kimyasal özellikleri gibi konularda kritik bilgiler elde edilir.

Özellikle biyolojik sistemlerde, MD simülasyonları proteinlerin katlanma süreçlerini, enzim-substrat etkileşimlerini ve hücre zarı dinamiklerini incelemek için kullanılır. İlaç tasarımı alanında, potansiyel ilaç adaylarının hedef proteinlere nasıl bağlandığı ve bu bağlanmanın moleküler düzeyde nasıl gerçekleştiği gibi sorulara yanıt bulunmasına yardımcı olur. Malzeme biliminde ise polimerlerin, metallerin ve kompozit malzemelerin mekanik, termal ve elektronik özelliklerinin anlaşılması için MD simülasyonlarından faydalanılır. Ayrıca, gaz depolama, kataliz ve ayırma teknolojileri gibi uygulamalarda kullanılan metal-organik çerçeveler gibi malzemelerin davranışları da bu yöntemle incelenebilir.

MD simülasyonlarının gücü, gerçekçi fiziksel koşullar altında moleküler sistemlerin dinamiklerini yüksek hassasiyetle modelleyebilmesidir. Bu, deneysel olarak gözlemlenmesi zor veya imkânsız olan atomik düzeydeki süreçlerin anlaşılmasına olanak sağlar. Bununla birlikte, MD simülasyonlarının başarısı, kullanılan kuvvet alanlarının doğruluğuna, hesaplama gücüne ve simülasyon süresine bağlıdır. Günümüzde yüksek performanslı bilgisayarlar ve gelişmiş algoritmalar sayesinde, daha büyük ve karmaşık sistemler daha uzun süreler boyunca simüle edilebilmekte, bu da bilimsel keşiflerin hızlanmasına katkıda bulunmaktadır. Sonuç olarak, moleküler dinamik simülasyonları, modern bilim ve mühendislikte vazgeçilmez bir araç haline

gelmiştir. Makromoleküler yapının incelenmesi biyolojinin anlaşılmasında önemli bir noktadır. Biyolojik işlev moleküler etkileşimlere dayanır ve bunlar makromoleküler yapıların bir sonucudur. 1950'li yıllardaki ilk yapı belirlemelerinden bu yana hem protein hem de nükleik asit dünyasında, makromoleküler yapıların nasıl inşa edildiğine dair bilgi sürekli olarak artmaktadır. Şu anda, protein veri bankası (PDB), 100.000'den fazla protein, 2.800 nükleik asit, tek başına veya kompleksler oluşturarak yaklaşık 20.000 küçük molekülün makromoleküllerle kompleks oluşturması dahil olmak üzere 110.000'den fazla giriş verisi içermektedir (Berman, H. M., et al, 2000).

Bu tür yapısal bilgi tarafından tanımlanan moleküler tanıma kuralları, enzim mekanizmaları ve düzenlemesi, zarlar boyunca taşıma, ribozomlar veya viral kapsidler gibi büyük yapıların oluşturulması veya DNA'nın nasıl okunduğu ve transkripsiyonun nasıl kontrol edildiği gibi temel biyolojik olayların anlaşılmasını sağlar. Protein-protein etkileşim ağlarının incelenmesi ve tahmini, modern sistem biyolojisinde büyüyen alanlardan biridir. Daha pratik anlamda, protein üç boyutlu (3B) yapıları ve yapıya dayalı ilaç tasarımının temelini oluşturur (Peluso, P., and Chankvetadze, B. ,2022).

GROMACS, MD simülasyonları için yaygın olarak kullanılan bir yazılım paketidir; ancak girdi dosyalarının hazırlanması ve sistem kurulumu karmaşık ve zaman alıcı olabilmektedir. Bu çalışmada, GROMACS tabanlı MD simülasyonlarının hazırlanmasını kolaylaştırmayı ve basitleştirmeyi amaçlayan özel bir araç geliştirilmiştir. Bu çalışma sınırlı hesaplama deneyimine sahip araştırmacıları simülasyonları verimli bir şekilde kurma ve yapılandırma konusunda güçlendirmek ve moleküler dinamik çalışmaları yürütmenin önündeki engelleri azaltmak için tasarlanmıştır.

GROMACS tabanlı MD simülasyonlarını hazırlamak için kullanıcı dostu ve verimli bir yazılım aracı oluşturmak ve GROMACS sistemi kurulumunda yer alan karmaşık ve hataya açık adımların çoğunu otomatikleştirerek, farklı düzeylerde hesaplama uzmanlığına sahip araştırmacıların erişebilmesini sağlamak amaçlanmaktadır. Ayrıca, GROMACS performansını daha da artırmak ve kaynakları verimli kullanmak için kuyruk sistemi oluşturarak bir çalışma bittiğinde sıradaki projeye geçecek şekilde

tasarlanmıştır. Sistem üzerinden grup oluşturarak aynı ya da benzer çalışmaların ortak grafiklerinin oluşturulması sağlanmıştır.

Sonuç olarak, bu çalışma araştırmacıları, öğrencileri ve eğitimcileri arayüz sayesinde UNIX Terminali kullanmadan analizlerini yapıp sonuçlarını izleyebilecekler. Bulut platformlarında GROMACS kullanmanın doğasında bulunan sınırlamaları ele alarak, eğitim amaçlı kullanım, yenilikçi keşifler ve çeşitli bilimsel alanlardaki ilerlemeler için moleküler dinamik araştırmalarında daha fazla erişilebilirlik ve verimlilik sağlanması amaçlanmıştır.

Bu tez çalışması beş bölümden oluşmaktadır: Birinci bölümde moleküler dinamik simülasyonlarının temelleri ve web tabanlı yönetim sistemlerinin önemi tartışılmıştır. İkinci bölümde teorik altyapı ve literatür taraması sunulmuştur. Üçüncü bölümde GROMACS, Laravel ve Python entegrasyonu ile geliştirilen metodoloji detaylandırılmış, sistem mimarisi ve test protokolleri açıklanmıştır. Dördüncü bölümde protein-ligand kompleksleri ve apoproteinleri üzerinde yapılan test sonuçları geleneksel yöntemlerle karşılaştırmalı analizler paylaşılmıştır. Beşinci bölümde ise bulgular değerlendirilmiş, makine öğrenmesi entegrasyonu ve bulut bilişim desteği gibi gelecek çalışma önerileri sunularak platformun biyoinformatik ve ilaç tasarımındaki potansiyel etkisi vurgulanmıştır.

2. LİTERATÜR ÖZETİ

Moleküler dinamik simülasyon, atomik düzeyde moleküllerin hareketlerini simüle etmek için kullanılan bir hesaplama yöntemidir. Bu yöntem, Newton'un hareket denklemlerine dayanarak, moleküler etkileşim kuvvetlerini ve parçacıkların hareketini sayısal olarak hesaplayarak moleküler sistemlerin davranışını incelemektedir. Simülasyon süreci boyunca, sistemdeki her atomun konumu ve hızı zamanla değişir, bu da moleküler sistemlerin dinamik davranışlarını gözlemlememizi sağlar. Ayrıca, simülasyon sonuçları, sistemdeki termodinamik ve yapısal değişikliklerin yanı sıra, moleküler etkileşimlerin ve reaksiyonların anlaşılmasına yönelik değerli bilgiler sağlayabilmektedir.

2.1 Moleküler Dinamik Simülasyon Temelleri

Moleküler dinamik simülasyonları, atomların ve moleküllerin hareketlerini modellemek için kuvvet alanlarına (force fields) dayanır. Kuvvet alanları, atomlar arasındaki bağ uzunlukları, açılar, dihedral açılar ve elektrostatik/van der Waals etkileşimlerini tanımlayan matematiksel modellerdir. Bu modeller, sistemin potansiyel enerjisini hesaplar ve atomların hareketlerini belirler. Popüler kuvvet alanları arasında AMBER, CHARMM ve OPLS bulunmaktadır (Frenkel and Smit, 2002).

Simülasyonlar sırasında, atomların hareket denklemleri sayısal olarak çözülür. Bu işlem için Verlet, Leapfrog veya Velocity-Verlet gibi entegrasyon algoritmaları kullanılır. Bu algoritmalar, enerji korunumu ve hesaplama verimliliği sağlamak için optimize edilmiştir. Ayrıca, simülasyonlar sırasında periyodik sınır koşulları uygulanarak sistemin sonsuz bir ortamda simüle edilmesi sağlanır. Bu yöntem, yüzey etkilerini minimize eder ve gerçekçi sonuçlar elde edilmesine yardımcı olur (Allen and Tildesley, 2017).

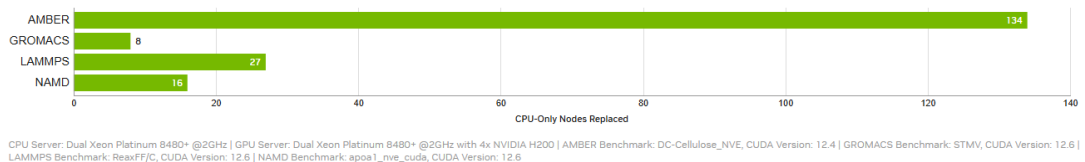
Moleküler dinamik simülasyonları için birçok yazılım geliştirilmiştir. Bu yazılımlar, farklı kullanım alanlarına ve hesaplama gereksinimlerine göre optimize edilmiştir. Örneğin, GROMACS, yüksek hızı ve GPU desteği ile bilinir ve özellikle protein katlanması ve lipid membranlar gibi biyolojik sistemlerin simülasyonlarında yaygın olarak kullanılır (Abraham et al., 2015). AMBER ise biomoleküller için optimize edilmiş bir yazılımdır ve DNA/RNA dinamikleri ile ilaç tasarımı gibi alanlarda sıklıkla tercih edilir (Case et al., 2021).

Büyük ölçekli sistemlerin simülasyonu için NAMD oldukça popülerdir. Paralel hesaplama yetenekleri sayesinde viral kapsidler ve hücre zarı proteinleri gibi karmaşık sistemlerin modellenmesinde kullanılır (Phillips et al., 2020). Malzeme bilimi alanında ise LAMMPS öne çıkar. Bu yazılım, metaller, polimerler ve nano yapılar gibi malzemelerin mekanik özelliklerini incelemek için kullanılır (Plimpton, 1995). Son olarak, OpenMM, GPU optimizasyonu ve Python API'si ile esnek simülasyon protokolleri sunar ve özellikle yeni başlayanlar için kullanımı kolay bir seçenektir (Eastman et al., 2017).

2.2 GROMACS ve Diğer MD Yazılımlarının Karşılaştırması

GROMACS, NAMD, Maestro-Schrödinger, AMBER ve CHARMM gibi moleküler dinamik yazılımları karşılaştırmak önemlidir. GROMACS, açık kaynak yazılım birçok platformda kullanılabilir ve paralel hesaplama özelliklerine sahiptir. NAMD ve OpenMM, büyük sistemlerde yüksek performans gösterir ve paralel hesaplama yetenekleri vardır. Maestro-Schrödinger, moleküllerin tasarımı ve simülasyonu için kapsamlı bir platform sağlar. AMBER, güçlü termodinamik hesaplamalar için kullanılırken, CHARMM, proteinler ve küçük moleküllerin simülasyonları için idealdir. Bu yazılımların karşılaştırılması, hangi durumlarda hangi yazılımın daha uygun olduğunu anlamak için önemlidir.

Molecular Dynamics



Şekil 2.1 NVIDIA firmasının developer NRF karşılaştırma grafiği (<https://developer.nvidia.com>, 2025)

Şekil 2.1’de NVIDIA firmasının yapmış olduğu hız testlerinin h200 modeli ile kaç çekirdek kullanıldığını gösteren bir test gerçekleştirilmiştir. GROMACS ücretli sürümlere göre iyi bir performans göstermiştir. MD simülasyon yazılımları önemli özellikleri bakımından karşılaştırmaları yapmak için Çizelge 2.1 hazırlanmıştır.

Çizelge 2.1 MD yazılımlarının bazı özellikleri bakımından karşılaştırılması

Özellik	GROMACS	AMBER	NAMD	OpenMM	LAMMPS	CHARMM	Maestro-Schrödinger
Kullanım Alanı	Biyo-moleküller	Biyo-moleküller	Büyük sistemler	Esnek simülasyonlar	Malzeme bilimi	Biyo-moleküller	İlaç keşfi
GPU Desteği	Evet	Evet	Evet	Evet	Evet	Evet	Evet
Paralel Hesaplama	Yüksek	Orta	Yüksek	Orta	Yüksek	Orta	Orta
Kullanım Kolaylığı	Orta	Yüksek	Orta	Yüksek	Düşük	Orta	Yüksek
Açık Kaynak	Evet	Hayır	Evet	Evet	Evet	Hayır	Hayır

NVIDIA tarafından geliştirilen GPU mimarilerinin performans karşılaştırması, özellikle moleküler dinamik (MD) simülasyonlarında kullanılan hesaplama kapasiteleri açısından Şekil 2.2’de sunulmuştur. Bu karşılaştırmada, GPU’ların CUDA Core sayıları ve hesaplama verimlilikleri detaylı olarak incelenmiştir. Performans metriklerinden biri olan "ns/gün" (nanosaniye/gün), birim zamanda simüle edilebilen sistem süresini ifade eder ve bu değerin yüksekliği, simülasyonun daha uzun zaman dilimlerini kapsayabildiğini göstererek hesaplama performansının bir göstergesidir. Bu parametre, özellikle GROMACS gibi yüksek performans gerektiren moleküler dinamik yazılımlarında sistem verimliliğinin değerlendirilmesinde kritik bir rol oynamaktadır.

App Name	System Name	RX 7900XTX	RX 6900XT	Radeon VII	RTX 4090	RTX 4080	RTX 4070Ti	RTX 4070	RTX 4060Ti	RTX 4060	RTX 3080Ti OC	ARC A770
GROMACS (ns/day)	A	58.7580	48.1640	26.0020	137.3720	100.7570	78.5890	62.1140	46.6570	32.8750	71.7050	27.8600
	A-2	74.6160	63.0200	44.1410	173.2880	131.1490	114.1220	94.9690	73.8570	55.1220	103.8080	37.5870
	B	131.9890	118.7030	73.8120	353.3360	273.6980	212.3230	179.3310	136.1260	97.8860	203.5940	84.2090
	B-Ti	106.0820	88.6590	55.2330	242.2330	201.0160	163.2880	130.6750	105.2210	76.8440	121.5510	54.2880
	STMV-NPT	20.1347	16.1955	8.4435	43.5167	32.4846	24.8407	19.5188	14.4238	10.2387	21.5665	9.2133
	STMV-GMX	23.2884	18.0518	8.8209	48.1039	33.7955	25.3898	19.8497	14.6163	10.3854	22.3545	10.0482
	benchPEP-h	2.1188	1.6265	0.8763	4.3498	3.0319	2.2965	1.8551	1.3393	0.9840	2.4000	1.1058
GROMACS SYCL + CUDA (ns/day)	A				115.5830	83.7400	65.5480	52.2600	39.3340	27.9940	59.1800	
	A-2				151.5880	114.8980	96.9510	81.1600	63.6100	48.2040	87.8650	
	B				302.7170	233.1350	186.9630	153.3800	120.0650	84.5090	171.3040	
	B-Ti				206.9630	175.4640	142.3570	116.3310	93.2890	67.6910	107.6170	
	STMV-NPT				38.3460	27.6800	21.1118	16.8684	12.4621	8.9170	18.5132	
	STMV-GMX				42.1064	28.7840	21.5360	17.1293	12.6616	8.9923	19.1301	
	benchPEP-h				8.4828	6.8433	5.2665	4.5895	3.2097	2.6612	6.2268	
AMBER (ns/day)	A	46.15	35.20	22.29	68.23	47.82	36.02	29.10	21.96	15.59	32.47	
	A-2	70.90	54.76	35.72	97.43	70.39	53.39	44.30	33.40	24.75	49.05	
	B	128.46	100.20	58.57	183.94	140.46	108.05	88.86	70.08	50.10	97.29	
	STMV-AMBER	51.06	36.75	26.14	74.83	49.36	35.45	28.85	19.39	15.34	35.93	
	STMV-AMBER-boost	56.23	39.92	28.49	80.86	53.36	38.22	31.02	20.77	16.47	38.71	
	STMV-NPT	18.37	13.37	8.60	28.80	19.48	13.91	11.14	7.96	5.83	12.71	
	STMV-OpenMM	45.09	32.85	23.68	61.69	42.14	28.97	23.53	15.75	12.51	27.77	
	Cellulose-AMBER	136.62	84.16	51.11	205.30	145.74	110.77	90.32	65.70	46.91	100.78	
	Cellulose-AMBER-boost	146.92	89.56	53.54	220.52	156.48	118.05	95.85	69.61	49.48	107.17	
	Cellulose-OpenMM	121.75	77.79	48.69	172.97	124.45	92.45	76.26	56.35	40.41	85.23	
OpenMM (ns/day)	gbsa	2336.9667	1952.9800	1214.8200	3813.9867	3671.7533	3067.9300	2744.9433	2475.2900	1973.8700	2749.8133	
	rf	1865.6533	1611.0800	998.1850	2638.2200	2525.5167	2197.9767	1958.6767	1652.7833	1241.3267	1866.0600	
	pme	1545.7333	1406.1367	859.4480	2250.0100	2104.8267	1765.6400	1514.3367	1333.2667	940.4230	1583.7600	
	apoa1rf	706.1797	576.1043	314.8150	1179.6533	950.5740	768.9767	607.7677	486.2560	328.8220	666.7433	
	apoa1pme	605.5960	496.5410	267.2130	784.5597	627.5333	499.5570	386.7357	314.5340	222.2083	449.1547	
	apoa1lpme	503.8287	405.4043	217.6460	620.8740	496.3347	395.9520	306.4817	251.1783	177.7993	360.3853	
	amoebapme	26.3503	17.9901	9.7990	44.3153	37.4695	28.8937	23.8067	20.4750	14.3373	28.4805	
	amber20-cellulose	154.3160	118.1527	62.9543	201.1663	160.3793	118.0797	87.4493	68.3161	47.1597	81.2174	
	amber20-stmv	51.0472	39.0811	22.6515	59.3497	43.6039	28.1131	20.3838	14.9091	11.9338	23.5154	
	LI 2.5	1.9846E+08	2.2934E+08	1.8125E+08	2.9912E+08	1.8195E+08	1.3777E+08	1.1106E+08	8.0554E+07	5.4123E+07	1.3665E+08	
LAMMPS (atom-step/s)	EAM	6.4274E+07	7.5025E+07	6.6283E+06	8.9636E+07	5.4404E+07	4.1337E+07	3.3195E+07	2.4212E+07	1.7170E+07	4.0343E+07	
	ReaxFF/C	9.7861E+05	1.0520E+06	1.1310E+06	1.4490E+06	8.5912E+05	6.4058E+05	5.2698E+05	3.8964E+05	2.6083E+05	6.4372E+05	
	Tersoff	9.8985E+07	1.2104E+08	8.5349E+07	1.2999E+08	7.8656E+07	5.9500E+07	4.7749E+07	3.4820E+07	2.4548E+07	5.8667E+07	

Şekil 2.2 NVIDIA GPU ve CUDA Core karşılaştırmaları (<https://developer.nvidia.com,> 2025)

2.2.1 GROMACS

GROMACS, özellikle biyomoleküler sistemlerin simülasyonu için optimize edilmiş, yüksek performanslı bir MD yazılımıdır. Açık kaynak kodlu olması ve GPU desteği sayesinde hem akademik hem de endüstriyel kullanıcılar arasında oldukça popülerdir. GROMACS, özellikle protein katlanması, lipid membranlar ve solvasyon sistemleri gibi biyolojik süreçlerin modellenmesinde sıklıkla kullanılır. Yazılım, hız ve hesaplama verimliliği açısından öne çıkmakta ve büyük ölçekli sistemlerde bile yüksek performans sunabilmektedir (Abraham et al., 2015). Şekil 2.3’de GROMACS son logosu gösterilmektedir.



Şekil 2.3 GROMACS logosu (<https://gromacs.org>, 2025)

GROMACS sistemleri, moleküler dinamik simülasyonlarını gerçekleştirmek için kullanıcılara erişilebilmekte, esnek ve güçlü bir platform sunmaktadır. Özellikle yüksek performanslı hesaplama kaynaklarına erişimi olmayan kullanıcılar için bu sistemler, karmaşık simülasyonları gerçekleştirme imkânı sağlamaktadır. Ancak, online sistemlerin sunduğu bu avantajlar, beraberinde bazı zorlukları da getirmektedir. İnternet bağlantısı sorunları, web arayüzünün kullanım zorlukları, sunucu tarafındaki kısıtlamalar ve veri güvenliği endişeleri, online GROMACS kullanımını etkileyen önemli faktörler arasında yer almaktadır.

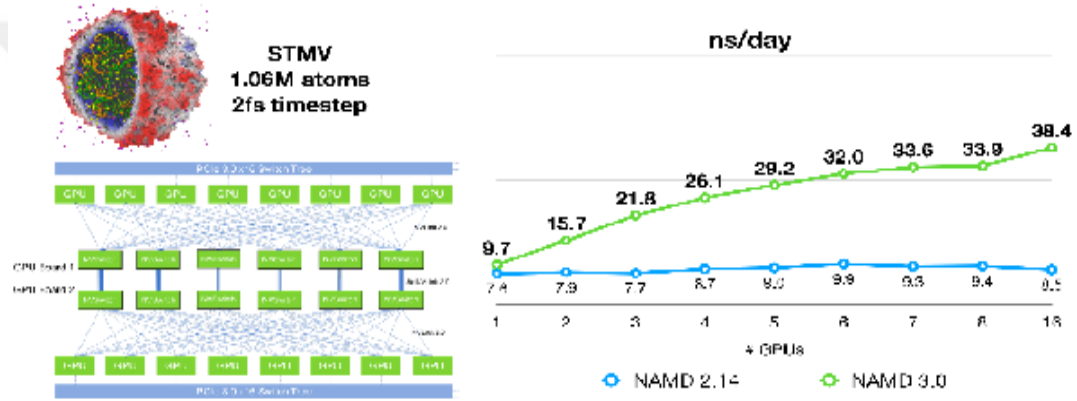
Bu zorlukların üstesinden gelmek için, kullanıcıların stabil bir internet bağlantısı kullanması, platformun sunduğu belgelendirme ve destek kaynaklarından yararlanması ve verilerini düzenli olarak yedeklemesi önerilmektedir. Ayrıca, online GROMACS platformlarının geliştiricileri tarafından kullanıcı dostu arayüzlerin tasarlanması, hata mesajlarının daha açıklayıcı hale getirilmesi ve sunucu altyapısının güçlendirilmesi, bu sistemlerin daha verimli ve güvenilir bir şekilde kullanılmasını sağlayacaktır.

Sonuç olarak, online GROMACS sistemleri, moleküler dinamik alanında çalışan araştırmacılar için önemli bir araç olmaya devam etmektedir. Ancak, bu sistemlerin sunduğu potansiyelin tam olarak kullanılabilmesi için hem kullanıcıların hem de platform geliştiricilerinin ortak çaba göstermesi gerekmektedir. Gelecekte, internet altyapısının iyileştirilmesi ve yapay zekâ destekli optimizasyonların devreye alınmasıyla, online GROMACS sistemlerinin daha da gelişerek, bilimsel araştırmalara daha büyük katkılar sağlayacağı öngörülmektedir.

2.2.2 NAMD

NAMD, büyük ölçekli sistemlerin simülasyonu için tasarlanmış bir MD yazılımıdır. Paralel hesaplama yetenekleri sayesinde viral kapsidler, hücre zarı proteinleri ve diğer karmaşık biyolojik sistemlerin modellenmesinde sıklıkla kullanılır. NAMD, GROMACS'a kıyasla daha büyük sistemlerde daha iyi performans gösterir, ancak küçük ve orta ölçekli sistemlerde GROMACS daha hızlı olabilmektedir. (Phillips et al., 2020).

NAMD 3.0: Single trajectory - Multiple GPU Performance



Şekil 2.4 NAMD 2.14 ile 3.0 versiyonu GPU performans karşılaştırması (<https://www.ks.uiuc.edu>, 2025)

Şekil 2.4’de görüldüğü gibi GPU kullanımı MD simülasyonlarında büyük bir gelişme yaşanmasını sağlamaktadır. Hızlı hesaplama kapasiteleri sayesinde büyük ölçekli simülasyonlar kolayca yapılabilmektedir.

2.2.3 Maestro-Schrödinger

Maestro-Schrödinger, ilaç keşfi ve moleküler modelleme alanlarında yaygın olarak kullanılan ticari bir yazılım paketidir. GROMACS ve diğer açık kaynaklı yazılımlardan farklı olarak, Maestro-Schrödinger, kullanıcı dostu grafiksel arayüzü (GUI) ve entegre araçları ile öne çıkmaktadır. Özellikle protein-ligand etkileşimleri, sanal tarama ve ilaç

adayı optimizasyonu gibi süreçlerde sıklıkla kullanılır. Ancak, ticari bir yazılım olması nedeniyle lisans maliyetleri yüksektir ve bu durum, özellikle akademik kullanıcılar için bir sınırlama oluşturabilmektedir. Ayrıca, GROMACS ve NAMD gibi yazılımlara kıyasla büyük ölçekli sistemlerde performansı daha düşük olabilmektedir. (Schrödinger, 2023).

2.2.4 AMBER

AMBER, biyomoleküler sistemler için özel olarak tasarlanmış bir MD yazılımıdır. Özellikle DNA/RNA dinamikleri, protein-ligand etkileşimleri ve ilaç tasarımı gibi alanlarda yaygın olarak kullanılmaktadır. AMBER, kullanıcı dostu arayüzü ve geniş kuvvet alanı seçenekleri ile bilinir. Ancak, GROMACS'a kıyasla daha az paralel hesaplama desteği sunmakta ve büyük sistemlerde performansı daha düşük olabilmektedir (Case et al., 2021).

2.2.5 OpenMM

OpenMM, esnek simülasyon protokolleri ve GPU optimizasyonu ile öne çıkan bir MD yazılımıdır. Python API'si sayesinde kullanıcılar, özelleştirilmiş simülasyonlar kolayca tasarlayabilir. OpenMM, özellikle yeni başlayanlar ve özelleştirilmiş simülasyonlar üzerinde çalışan araştırmacılar için idealdir. Ancak, GROMACS'a kıyasla daha az kullanıcı dostu bir arayüze sahiptir ve büyük ölçekli sistemlerde performansı daha düşük olabilmektedir (Eastman et al., 2023).

2.2.6 CHARMM

CHARMM (Chemistry at HARvard Macromolecular Mechanics), biyomoleküler sistemlerin simülasyonu için geliştirilmiş bir MD yazılımıdır. Özellikle proteinler, nükleik asitler ve lipidler gibi biyolojik makromoleküllerin dinamiklerini incelemek için kullanılır. CHARMM, kendi kuvvet alanı ile ünlüdür ve bu alan, biyomoleküler sistemler için oldukça doğru sonuçlar sunar. Ancak, GROMACS'a kıyasla daha az

kullanıcı dostu bir arayüze sahiptir ve büyük sistemlerde performansı daha düşük olabilmektedir (Brooks et al., 2009).

2.3 Simülasyon Otomasyonu: Mevcut Yaklaşımlar

Moleküler dinamik (MD) simülasyonunda otomasyon, süreçlerin verimliliğini artırmak ve kullanıcı hatalarını en aza indirmek açısından büyük önem taşır. Bu nedenle, mevcut otomasyon yaklaşımlarının incelenmesi ve değerlendirilmesi, simülasyon süreçlerinin optimize edilmesi için kritik bir adımdır. Günümüzde, MD simülasyonlarında otomasyon genellikle Python veya Shell script gibi programlama dilleri kullanılarak gerçekleştirilmektedir. Bu yaklaşımlar, kullanıcıların ihtiyaçlarına göre esnek bir şekilde özelleştirilebilmekte ve simülasyon süreçlerinin farklı aşamalarını otomatikleştirmek için kullanılabilmektedir.

Otomasyon araçları, özellikle NAMD veya GROMACS gibi popüler MD yazılımlarının giriş dosyalarını oluşturmak, simülasyonu çalıştırmak ve elde edilen sonuçları analiz etmek için yaygın olarak kullanılmaktadır. Örneğin, bash betikleri, simülasyonların otomatik olarak başlatılması, farklı parametrelerle çalıştırılması ve sonuçların otomatik olarak kaydedilmesi gibi işlemleri kolaylaştırır. Ayrıca, bu araçlar, büyük ölçekli simülasyon projelerinde veri yönetimi ve iş akışının yönetilmesi için de kullanılır. Bu sayede, karmaşık simülasyon süreçleri daha düzenli ve hatasız bir şekilde yürütülebilmektedir.

Ancak, mevcut otomasyon yaklaşımlarının hem avantajları hem de dezavantajları bulunmaktadır. Örneğin, Python ve Shell script gibi araçlar, esneklik ve özelleştirilebilirlik açısından büyük avantajlar sunmaktadır. Bununla birlikte, bu araçların kullanımı, programlama bilgisi gerektirdiğinden, özellikle biyoloji veya kimya gibi alanlardan gelen kullanıcılar için öğrenmede zorluklar oluşabilmektedir. Ayrıca, büyük ölçekli simülasyon projelerinde, otomasyon betiklerinin bakımı ve hata ayıklama süreçleri zaman alıcı olabilmektedir.

Bu bölümde, mevcut simülasyon otomasyon yaklaşımlarının avantajları ve dezavantajları detaylı bir şekilde değerlendirilerek, bu projenin ihtiyaçlarına en uygun yöntemin belirlenmesi hedeflenmektedir. Bu değerlendirme sürecinde, farklı otomasyon araçlarının performansı, kullanım kolaylığı, ölçeklenebilirlik ve entegrasyon yetenekleri gibi faktörler göz önünde bulundurulmuştur. Ayrıca, otomasyon süreçlerinin nasıl daha verimli hale getirilebileceği ve kullanıcı dostu çözümlerin nasıl geliştirilebileceği üzerine öneriler sunulmuştur.

Otomasyon araçları, örneğin bash betikleri, NAMD veya GROMACS gibi giriş dosyalarını oluşturmak, çalıştırmak ve sonuçları analiz etmek için sıklıkla kullanılır. Ayrıca, bu araçlar genellikle veri yönetimi ve iş akışının yönetilmesi için kullanılır. Bu bölümde, mevcut simülasyon otomasyon yaklaşımlarının avantajları ve dezavantajları detaylı şekilde değerlendirilerek, bu çalışmanın amacına uygun en doğru yöntemin belirlenmesi hedeflenmektedir.

2.4 Literatürde Yapılan Çalışmalar

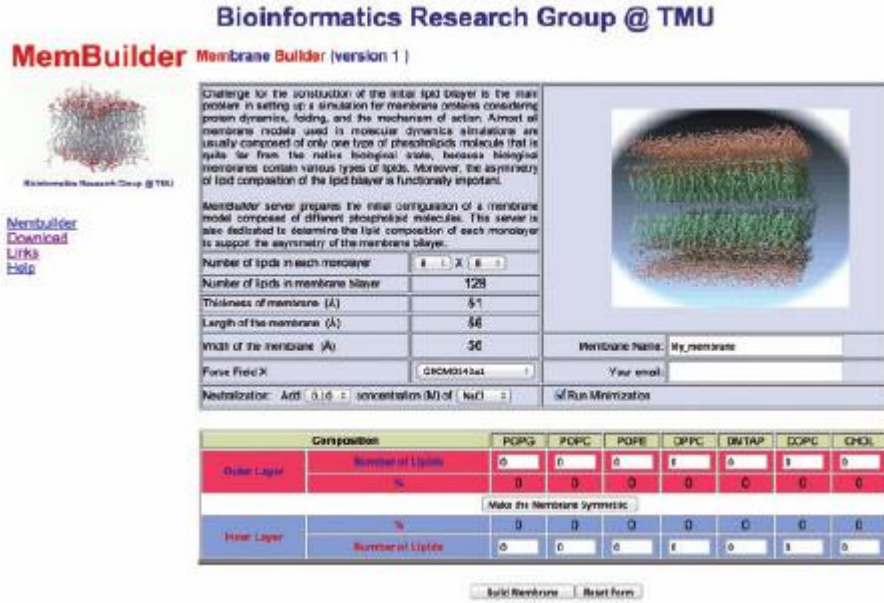
Moleküler dinamik simülasyonları, biyolojik makromoleküllerin yapı-fonksiyon ilişkilerini anlamada sağladığı değerli içgörüler nedeniyle giderek önem kazanmaktadır. Hesaplama gücündeki artış, simülasyon algoritmalarının optimizasyonu ve sonuçların daha hızlı elde edilmesi yönünde büyük bir talep yaratmıştır. Bununla birlikte, kullanım kolaylığı ihtiyacı da ön plandadır. Grafiksel kullanıcı arayüzü (GUI) programları, komut satırı tabanlı uygulamaların kullanımını kolaylaştıran önemli araçlar olarak öne çıkmaktadır. Komut satırı tabanlı programların etkin kullanımı, UNIX kabuğu ve en az bir UNIX tabanlı metin editörü hakkında temel bilgi gerektirir. Bu durum, biyologların bu tür araçları verimli bir şekilde kullanmasını zorlaştırabilir. Yaygın olarak kullanılan bir moleküler dinamik simülasyon ve analiz paketi olan GROMACS de bu konuda bir istisna değildir. Deneysel prosedürlerin temel bulgularını desteklemek için hesaplama yöntemlerine olan bağımlılığın artması, komut satırı tabanlı paketlerin daha kullanıcı dostu ve etkileşimli hale getirilmesi ihtiyacını artırmaktadır. Literatürde bu alanda yapılmış çalışmalar şunlardır:

Kota (2007) yaptığı çalışmada GROMACS'ın Linux sürümü (sürüm 3.3) için Java tabanlı kullanıcı etkileşimli bir ön uç arayüzü olan GUIMACS'ı sunmuştur. GUIMACS, kullanıcılarının aynı anda birden fazla moleküler dinamik simülasyonunu çalıştırmasını ve/veya analiz etmesini sağlayan Çoklu Belge Arayüzü (MDI) ile bağımsız bir uygulama olarak çalışmaktadır (Kota, P.,2007).

Sellis ve diğerleri (2009) yaptıkları Gromita adlı çalışma yakın zamanda güncellenen moleküler dinamik paketi GROMACS, sürüm 4'e tamamen entegre ve etkili bir grafiksel kullanıcı arayüzüdür. Gromita, komut satırı engelini aşmak ve GROMACS aracılığıyla moleküler dinamik simülasyonlarını çalıştırmak için yeni bir kullanıcı dostu ortam sunmak üzere tasarlanmış, çapraz platformlu, perl/tcl-tk tabanlı, etkileşimli bir ön uçtur (Sellis et al, 2009).

Hospital ve diğerleri (2012) yaptıkları MDWeb ve MDMoby adlı çalışmalar Amber, NAMD ve GROMACS'ı entegre eden bir platform özelliği göstermektedir ve standart simülasyon protokolleri için hazır şablonlar içermektedir. BioExcel projesi kapsamında yapılan çalışmalar ise GROMACS'ın API altyapısını genişleterek heterojen paralelleştirme ve bulut bilişim entegrasyonu gibi yenilikler eklemektedir (Hospital et al,2012).

Ghahremanpour ve diğerleri (2014) heterojen lipid tabakaları oluşturmak için tasarlanmış bir web tabanlı araç olan MemBuilder adlı çalışmayı sunmuşlardır. Sunulan bu platform, GROMACS topoloji dosyalarını otomatik olarak oluşturmakta ve asimetrik lipid bileşimlerini destekleyerek enerji minimizasyonu yapılmış başlangıç koordinatlarını sağlamaktadır. Koordinat giriş yapılan arayüz Şekil 2.2.5'te gösterilmiştir. (Ghahremanpour et al,2014)



Şekil 2.5 MemBuilder arayüzü (Ghahremanpour et al,2014).

Ribeiro ve diğerleri (2016) yaygın olarak kullanılan iki modelleme programı olan NAMD ve VMD'yi entegre ederek, yeni başlayanların ve uzmanların biyomedikal açıdan ilgili soruları ele almasını sağlayan, kullanıcı dostu bir yazılım olan QwikMD adlı çalışmayı yapmışlardır. Hem basit hem de gelişmiş MD simülasyonlarını etkileşimli olarak gerçekleştiren QwikMD, yaygın hataları kontrol eder ve simülasyonları hazırlamak, yürütmek ve analiz etmek için gerekli birçok adımı otomatikleştirmektedir (Ribeiro et al, 2016).

Heo ve Feig (2018) yaptıkları PREFMD (Protein Refinement via Molecular Dynamics) adlı çalışmada web sunucusu, protein yapılarını moleküler dinamik simülasyonları ile rafine etmeye yönelik bir platform oluşturmuşlardır. Orijinal yöntem, 100 ns ölçekli MD simülasyonları sonrasında yapı seçimi ve topluluk ortalaması (ensemble averaging) adımlarını içermektedir. PREFMD web servisi, bu yöntemin doğruluğuna yakın sonuçlar verirken hesaplama maliyetlerini optimize edecek şekilde tasarlanmıştır (Heo ve Feig, 2018).

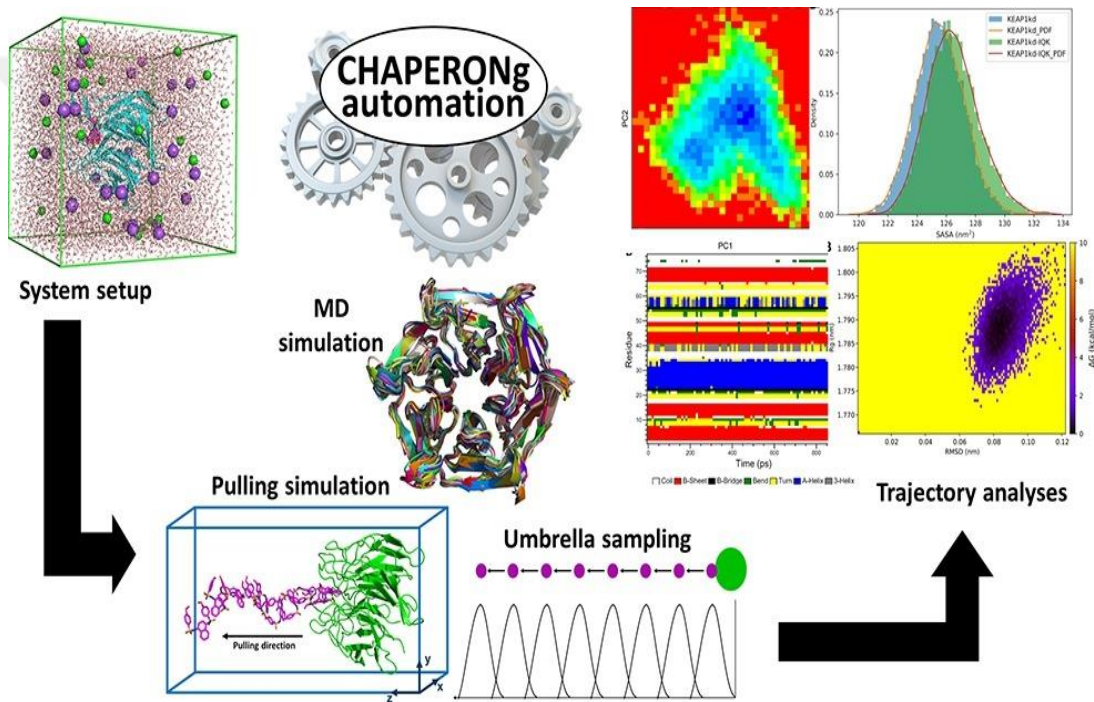
Rawat ve diğerleri (2021) HeroMDAnalysis adlı çalışmada bir kullanıcının farklı protein ve protein-ligand kompleks analizleri için farklı grafikler ve ham veriler oluşturmasına izin vermektedir, ancak GUI'yi kullanmak ve tüm parametreleri manuel

olarak belirtmek gerekmektedir. Öte yandan, MD simülasyonlarını analiz etmek için çok yararlı olan ancak komut dosyaları oluşturarak ve yürüterek kullanılabilen farklı işlevlere sahip araçlar veya paketler mevcuttur (Rawat R. et al,2021)

Sarkar ve diğerleri (2022) yaptıkları YAMACS adlı bir çalışma, YASARA moleküler görüntüleme yazılımı için bir eklenti paketi olarak GROMACS simülasyonlarını tamamen grafik arayüz üzerinden yönetmeyi sağlar. Bu araç, sistem hazırlama, simülasyon başlatma ve analiz adımlarını tek bir pencerede entegre ederken, gerçek zamanlı görselleştirme desteği sunarak kullanıcı deneyimini iyileştirir (Sarkar et al, 2022).

Vieira ve diğerleri (2023) yaptıkları VisualDynamics adlı çalışmada Flask tabanlı bir web uygulaması olarak GROMACS simülasyonlarını tarayıcı üzerinden yürütmeyi hedeflemişlerdir. Kullanıcı dostu arayüzü sayesinde, ligandlı ve ligandsız protein simülasyonlarını desteklerken analiz grafikleri ve log dosyalarını indirme imkanı sunmaktadır (Vieira et al,2023).

Yekeen ve diğerleri (2023) CHAPERONg adlı çalışma otomatik GROMACS tabanlı moleküler dinamik simülasyonları ve yörünge analizleri için bir araç olarak tanımlanan, Bash ve python ile oluşturulmuştur. MD simülasyonları yürütmek ve analiz etmek için yerleşik hatları kolaylaştıran bir çalışmadır. Yapısal olarak sunulan çalışmaya en yakın çalışmadır, fakat sadece otomatikleştirmeye yardımcı olmakta kullanabilmek için yine teknik bilgiye ihtiyaç duyulmaktadır. GitHub üzerinden devamlı güncellenmektedir (Yekeen et al, 2023).



Şekil 2.6 CHAPERONg çalışma mantığı (Yekeen et al,2023).

Park S.J. ve diğerleri (2025) sundukları çalışmada OPRLM yardımıyla simetrik veya asimetrik lipid dağılımlarına sahip 18 biyolojik zar tipine ve değişken kolesterol içeriğine sahip 5 tip iki bileşenli lipid çift katmanına karşılık gelen açık lipitlere sahip membran protein yapılarının birleştirilmesini otomatikleştirmişlerdir. CHARMM-GUI araç seti ve PPM yöntemi üzerine inşa edilen OPRLM, açık lipid karışımlarına sahip integral ve/veya periferik zar proteinlerini içeren karmaşık simülasyon sistemlerinin kurulumunu basitleştirir ve sonraki tüm atom MD simülasyonları için gerekli tüm dosyaları oluşturmaktadır (Park S.J. et al, 2025).

Bu tür web tabanlı çözümler, GROMACS'ın teknik karmaşıklığını azaltarak hem yeni başlayanlar hem de deneyimli kullanıcılar için simülasyon süreçlerini daha verimli hale getirmeyi amaçlamaktadır.

2.5 Python ve Shell Script Kullanımı Hakkında Temel Bilgiler

Ubuntu, Linux tabanlı bir işletim sistemi olarak, özellikle sunucu tarafında yaygın bir şekilde tercih edilmektedir. Shell scriptler, Ubuntu üzerinde otomasyon görevlerini gerçekleştirmek için kullanılan güçlü bir araçtır. Özellikle sistem yönetimi, dosya işlemleri ve süreç yönetimi gibi alanlarda shell scriptler, geliştiricilere hızlı ve etkili çözümler sunmaktadır. Python hem basit hem de karmaşık uygulamalar geliştirmek için kullanılan çok yönlü bir programlama dilidir. Veri analizi, makine öğrenimi, web geliştirme ve otomasyon gibi birçok alanda kullanılan Python'un esnek yapısı ve geniş kütüphane desteğine sahiptir.

Python, genellikle kodlama, betik yazma ve yazılım geliştirme için kullanılan bir programlama dilidir. Python betikleri, sistem yönetimi görevlerini otomatikleştirmek veya tekrarlanan görevleri kolaylaştırmak için kullanılabilir. Aynı şekilde, shell script'ler, sistem komutlarını sıralamak ve otomatikleştirmek için kullanılan metin dosyalarıdır. Python ve shell script'leri kullanarak, GROMACS tabanlı moleküler dinamik simülasyon hazırlığına yönelik arayüz geliştirme süreci sırasında simülasyon adımlarını otomatikleştirmek ve kullanıcıların iş yükünü azaltmak mümkündür. Her iki aracın da temel kullanım bilgileri uygun bir şekilde öğrenilmeli ve uygulanmalıdır.

2.6 PHP-Laravel ve MySQL kullanımı Hakkında Temel Bilgiler

PHP Laravel, modern web uygulamaları geliştirmek için kullanılan popüler bir framework'tür. Laravel, geliştiricilere temiz ve okunabilir kod yazma imkânı sunarken, aynı zamanda güçlü bir ORM (Object-Relational Mapping) yapısı olan Eloquent ile veritabanı işlemlerini kolaylaştırmaktadır. Laravel'in sunduğu blade template engine, routing yapısı ve artisan komut satırı aracı, web projelerinin hızlı ve etkili bir şekilde

geliştirilmesini sağlar. MySQL ise, açık kaynaklı bir ilişkisel veritabanı yönetim sistemi olarak, web uygulamalarının veri depolama ihtiyaçlarını karşılamak için yaygın olarak kullanılmaktadır. MySQL, yüksek performans, güvenilirlik ve kolay kullanımı ile öne çıkmaktadır. Laravel ile entegre bir şekilde çalışabilen MySQL, uygulamaların veri yönetimi ihtiyaçlarını karşılamak için ideal bir seçenektir.

Bu teknolojilerin bir arada kullanılması, özellikle full-stack web uygulamaları geliştirirken büyük avantajlar sağlamaktadır. Örneğin, Ubuntu üzerinde çalışan bir sunucuda, shell scriptler ile otomatik deployment işlemleri gerçekleştirilebilmektedir. Python, arka planda veri işleme veya API entegrasyonları için kullanılabilirken, Laravel ile dinamik ve kullanıcı dostu bir arayüz geliştirilebilmektedir. MySQL ise, uygulamanın veritabanı ihtiyaçlarını karşılayarak, verilerin güvenli ve tutarlı bir şekilde saklanmasını sağlamaktadır. Bu teknolojilerin birbirleriyle uyumlu çalışması, modern web uygulamalarının geliştirilmesinde önemli bir rol oynamaktadır.

2.7 Visual Molecular Dynamics (VMD) Programı

VMD, özellikle proteinler ve nükleik asitler gibi biyopolimerler olmak üzere moleküler yapıların görüntülenmesi ve analizi için tasarlanmış bir moleküler grafik programıdır. VMD, çok çeşitli işleme stilleri (çubuk, yüzey, ribbon vb.) ve renklendirme şemaları (atom tipi, yük, bölgeye özgü renklendirme) yöntemleri kullanarak aynı anda herhangi bir sayıda yapıyı görüntüleyebilmektedir. Moleküller, her bir gösterimin belirli bir işleme yöntemini ve atomların seçili bir alt kümesi için renklendirme şemasını içerdiği bir veya daha fazla "frame" olarak görüntülenir. Her gösterimde görüntülenen atomlar, Boole operatörleri ve düzenli ifadeler içeren kapsamlı bir atom seçimi sözdizimi kullanılarak seçilir. VMD, program kontrolü için eksiksiz bir grafik kullanıcı arayüzünün yanı sıra değişken ikamesi, kontrol döngüleri ve işlev çağrılarını içeren karmaşık betiklere izin vermek için TCL uzantılı, yerleştirilebilir ayrıştırıcıyı kullanan bir metin arayüzü sağlar. Daha sonra oynatılmak üzere bir VMD komut betiği üreten tam oturum kaydı desteklenir. Görüntülenen moleküllerin yüksek çözünürlüklü piksel tabanlı görüntüleri, gerçekçi görüntü dizisi görüntü işleme uygulaması tarafından kullanılmak üzere giriş betikleri üretilerek üretilebilir. VMD ayrıca, dosyalardan veya

alıřan bir MD simlasyonuna doėrudan baėlantıdan ie aktarılan molekler dinamik (MD) simlasyon yrngelerini canlandırma yeteneėiyle aıka tasarlanmıřtır. VMD, yapısal biyolojide etkileřimli problem zme araları seti olan MDScope'un grselleřtirme bileřenidir ve paralel MD programı NAMD ve grselleřtirme ve simlasyon programlarını baėlamak iin kullanılan MDCOMM yazılımını da iermektedir. VMD, nesne ynelimli bir tasarım kullanılarak C++'da yazılmıř, aık kaynak kodlu bir yazılımdır (Humphrey et al, 1996).

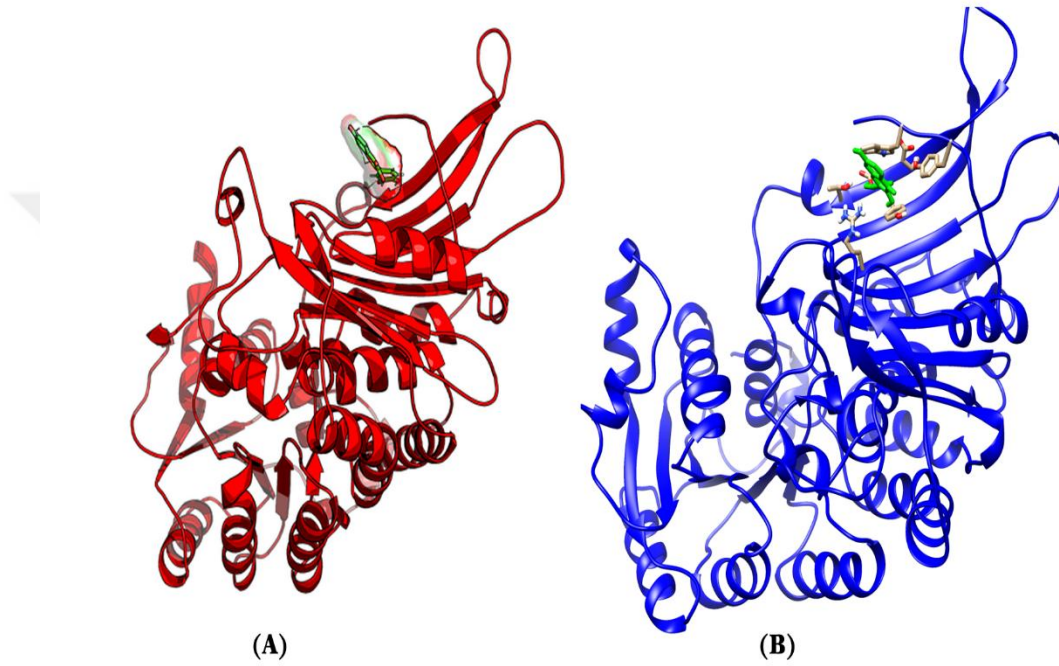
Bu alıřmada VMD, hidrojen baė oranları ve sayılarını hesaplamak iin kullanılmıřtır. TLC dosyası oluřturularak sistem zerinde kompleks ve simlasyon ile oluřan XTC dosyasını iřleyerek Hidrojen baėları ve oranlarını XVG olarak verdiėi ıktılar Python kodları ile grafiėe dnřtrlmektedir.

2.8 UCSF Chimera Programı

UCSF Chimera, molekler modelleme, grselleřtirme ve analiz iin kullanılan gl bir yazılımdır. zellikle biyomolekllerin (proteinler, DNA, RNA ve kompleks yapılar)  boyutlu modellenmesi ve analizi iin tasarlanmıřtır. UCSF Chimera, kullanıcıların molekler yapıları interaktif bir řekilde incelemelerine, farklı renklendirme ve grselleřtirme teknikleri kullanarak detaylı analizler yapmalarına olanak tanır (Pettersen et al., 2004). Program, X-ıřını kristalografisi, NMR spektroskopisi ve elektron mikroskopisi gibi deneysel yntemlerle elde edilen yapısal verilerin grselleřtirilmesinde yaygın olarak kullanılır.

UCSF Chimera, molekler dinamik (MD) simlasyonları, doku uyumu (docking) alıřmaları ve elektrostatik potansiyel hesaplamaları gibi hesaplamalı yntemlerle entegre alıřabilme zelliėine sahiptir. zellikle protein-ligand etkileřimlerinin analizi, hidrojen baėlarının belirlenmesi ve molekler yzeylerin incelenmesi gibi biyoinformatik ve yapısal biyoloji alıřmalarında sıklıkla tercih edilir. Ayrıca, programın eklenti (plugin) mimarisi sayesinde kullanıcılar, kendi ihtiyalarına zel analiz araları geliřtirebilir veya mevcut eklentileri kullanarak alıřmalarını geniřletebilir.

UCSF Chimera'nın bir diğ er  nemli  zelliđi, kullanıcı dostu aray z  ve detaylı dok mantasyonudur. Bu sayede hem yeni bařlayanlar hem de deneyimli kullanıcılar programı kolaylıkla  đrenebilir ve etkili bir řekilde kullanabilir. Program, akademik arařtırmacılar i in  cretsiz olarak sunulmakta ve s rekli g ncellenen  zellikleriyle bilimsel  alıřmalara  nemli katkılar sađlamaktadır (Pettersen et al., 2004).

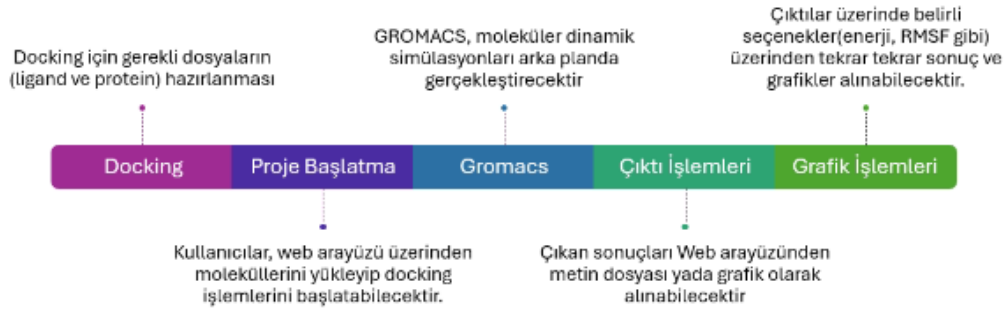


řekil 2.7 (A) Python ile (B) ise UCSF Chimera ile oluřturulmuř Protein-Ligand  rnek g rseli

Bu  alıřmada sim lasyon sırasında Protein ve Ligandın hareketlerini g rselleřtirmek i in bazı adımlar i in PDB formatında frame dosyaları oluřturulur. Bu dosyaları Python ile g rselleřtirmek řekil 2.8.1(A)'de g r nd đ  gibi bađlar tam anlařılır olmamakla beraber  ıktı oluřturmak i in sistem kaynaklarını yođun řekilde t ketmektedir Chimera ise oluřturduđumuz Python betiđini  alıřtırarak daha hızlı ve řekil 2.8.1(B)'de ki gibi g rsel olarak uygun  ıktılar vermektedir.

3. MATERYAL VE METOT

Yazılımın genel mimarisi, Python ve Shell Script entegrasyonu, Laravel-PHP ile web arayüz entegrasyonu, GROMACS komutlarının otomasyonu, geliştirme süreci, kodlama ve test aşamaları, algoritmalar ve akış diyagramları, kullanıcı arayüzü ve işlevsellik, yazılımın kullanıcı rehberi ve simülasyon protokolü gibi konular bu bölümde detaylı bir şekilde ele alınmıştır. Ayrıca hazırlık, enerji minimizasyonu, denge simülasyonları, üretim simülasyonu, analiz ve görselleştirme adımlarının nasıl gerçekleştirileceği, yazılımın performans testleri, işlem süresi ve verimlilik, hata oranı ve düzeltmeler gibi test sonuçları da bu bölümde yer almaktadır.



Şekil 3.1 Çalışmanın gelişme süreci

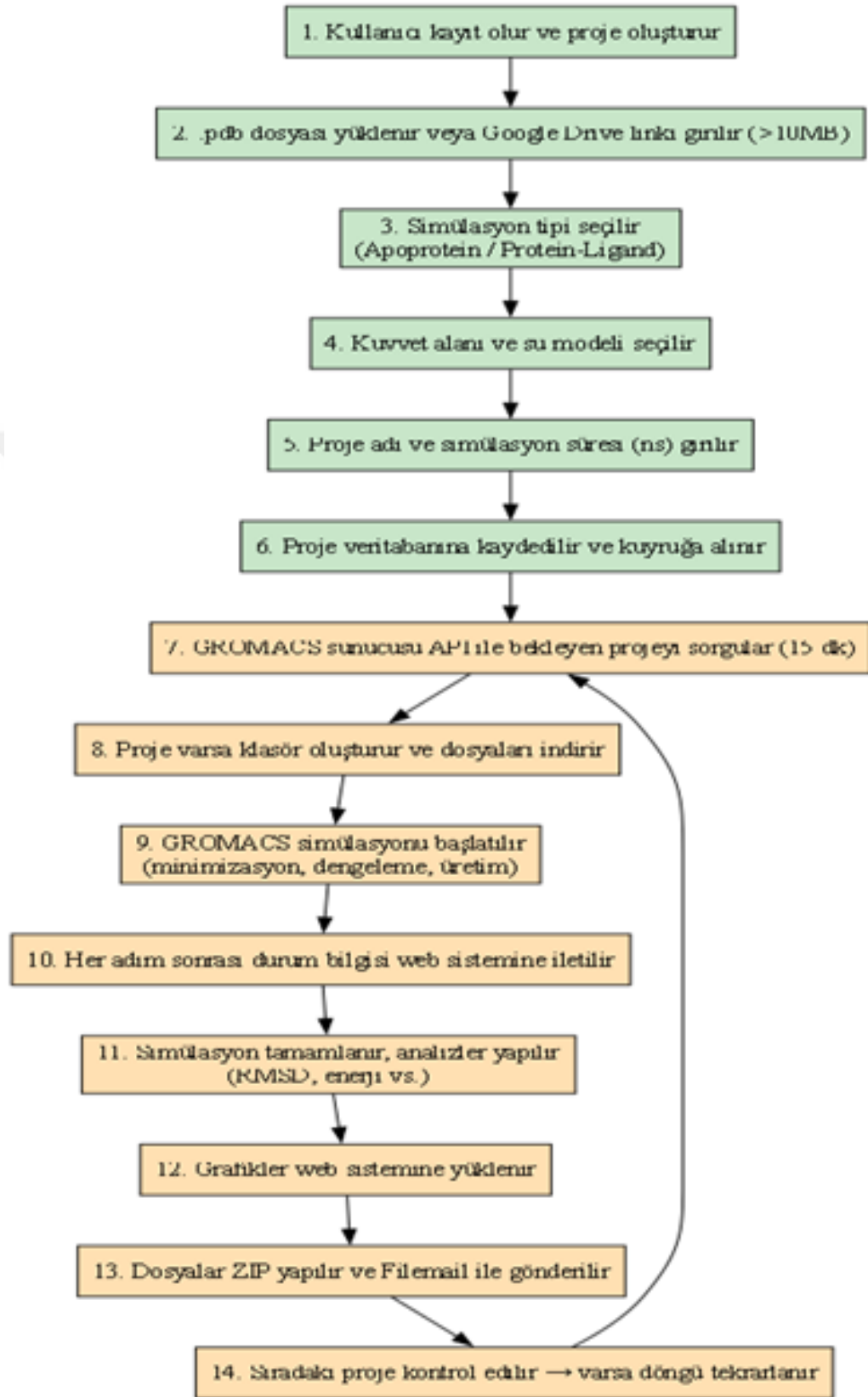
Çalışmayı tasarlarken şekil 3.1'deki MD simülasyon süreci, kullanıcıların ligand ve protein dosyalarını web arayüzüne yükleyerek projeyi başlatmasıyla başlamaktadır. Docking işlemi tamamlandıktan sonra GROMACS, arka planda moleküler dinamik simülasyonlarını otomatik olarak gerçekleştirmektedir. Elde edilen çıktılar üzerinde enerji, RMSF gibi parametrelerle analizler yapılır ve sonuçlar kullanıcıya metin dosyaları veya interaktif grafikler şeklinde sunulmaktadır. Bu adımlar, etkileşimlerin

atomik ölçekte incelenmesini ve veriye dayalı yorumlanmasını sağlayan sistematik bir işlem zincirini temsil etmektedir.

3.1 Yazılımın Genel Mimarisi

Yazılım birbirine servislerle bağlanan iki ana modül olarak oluşturulmuştur. Bunlardan birincisi Web arayüzü PHP/Laravel ve MySQL altyapısı kullanılarak oluşturulmuştur. Bu kısım bir web sunucusunda bulunacak ve diğer sistemler onunla iletişime geçerek gerekli iletişimi sağlamaktadır. Web tarafında, GROMACS sunucusunda devam eden ve biten işlerin listesi ve durumları saklanmaktadır. İkinci kısım GROMACS sunucusu, üzerinde Linux Shell komut dosyaları ve gerektiğinde kullanılacak Python dosyaları bulunmaktadır. Oluşturulan hazır komut betiğini bir Shell komutuna çevirerek kullanım kolaylaştırılmıştır. Her adımı takip edebilmek ve sonuç grafiklerini iletebilmek için web arayüzle iletişimi sağlanmıştır. Burada GROMACS sunucusu kısmı multifonksiyonel olabilecek şekilde tasarlanmıştır. Hatta GROMACS dışında MD simülasyon yazılımların da sisteme dahil edilmesine imkan sunmaktadır.

Moleküler dinamik (MD) simülasyon çalışmalarında sistematik bir çalışma akış şemasının izlenmesi, araştırmanın güvenilirliği ve tekrarlanabilirliği açısından kritik öneme sahiptir. Şekil 3.2’de MD simülasyonlarının standart protokollerine uygun olarak tasarlanmış olup, elde edilen sonuçların güvenilirliğini ve bilimsel geçerliliğini desteklemektedir. Çalışma akışının her aşaması, alanında kabul görmüş yöntemler ve yazılımlar kullanılarak gerçekleştirilmiştir.



Şekil 3.2 Çalışmanın akış şeması

3.2 Python ve Shell Script Entegrasyonu

Python ve Shell Script entegrasyonu, yazılım geliştirme ve sistem yönetimi süreçlerinde sıklıkla kullanılmaktadır. Bu entegrasyon, özellikle Unix/Linux tabanlı sistemlerde otomasyon, görev yönetimi ve veri işleme gibi işlemlerin daha etkili bir şekilde gerçekleştirilmesini sağlamaktadır. Python ve Shell Script'in birlikte kullanımı, bu iki dilin güçlü yönlerini birleştirerek daha verimli çözümler sunmaktadır.

Bu çalışmada, bir moleküler dinamik (MD) simülasyonu gerçekleştirmek için kullanılmıştır. Shell Script ile simülasyon dosyalarının hazırlanması, işlemlerin başlatılması ve sonuçların kaydedilmesi gibi adımlar otomatikleştirilmiştir. Python ile bu sonuçlar analiz edilip görselleştirilmiştir.

Python ile oluşturulan sistemde arka planda çalışan sistem Web arayüzü ile iletişimi sağlar ve projeyi başlatmak için Shell scripti tetikler. Shell script içerisinde yine web arayüzle ya da diğer bağlanması gereken sistemlerle iletişim için Python scriptleri kullanılmıştır. Görselleştirmeler için hem Python kütüphaneleri hem de yardımcı programlar kullanılmıştır.

3.3 GROMACS Komutlarının Otomasyonu

GROMACS, Linux Shell komutları ile kullanılmaktadır. Birçok komut yapısında farklı farklı parametre ve giriş/çıkış dosya yolları gerekmektedir. Bütün bu dosyaları oluşturacak ve sırayla çalışmaya hazır edecek komutlar oluşturulmuştur. Bu çalışmada, ilgili komutların yürütülmesini kolaylaştırmak amacıyla, tüm dosya yollarını ve parametreleri otomatik olarak tanımlayan bir komut seti geliştirilmiştir. Web arayüzü üzerinden kullanıcıdan alınan bilgiler doğrultusunda gerekli giriş dosyaları oluşturulmakta, ardından bu dosyalar kullanılarak enerji minimizasyonu, dengeleme (NVT/NPT) ve üretim (MD) aşamalarını kapsayan GROMACS komutları sırasıyla çalıştırılmak üzere hazırlanmakta ve tetiklenmektedir.

Böylece kullanıcı, GROMACS'ın karmaşık komut yapılarıyla doğrudan uğraşmadan, sadece temel parametreleri girerek simülasyon sürecini başlatabilmektedir. Bu otomasyon, hem kullanım kolaylığı sağlamakta hem de simülasyon sürecinin doğruluğunu ve tekrar edilebilirliğini artırmaktadır.

3.4 Geliştirme Süreci

Tez kapsamında sunulan yazılımın geliştirme süreci, projenin planlanmasından canlı ortama taşınmasına kadar birçok aşamadan oluşmaktadır. Bu süreç, web arayüzü için PHP-Laravel ve MySQL, arka plan işlemleri için ise Bash ve Python kullanılarak gerçekleştirilmiştir. Çalışmanın başarılı bir şekilde tamamlanması için her aşama titizlikle ele alınmış ve uygulanmıştır.

Çalışmanın ilk aşaması, ihtiyaç analizi ve planlama ile başlamıştır. Bu aşamada, çalışmanın temel amacı ve hedefleri belirlenmiş, kullanıcıların ihtiyaçları ve beklentileri analiz edilmiştir. Fonksiyonel ve fonksiyonel olmayan gereksinimler tanımlanarak, çalışmanın kapsamı netleştirilmiştir. Teknolojik altyapı olarak, web arayüzü için PHP-Laravel framework'ü ve MySQL veritabanı seçilmiş, arka plan işlemleri için ise Bash ve Python betiklerinin kullanılmasına karar verilmiştir.

Sistem tasarımı aşamasında, çalışmanın gereksinimlerine uygun bir mimari oluşturulmuştur. Laravel'in MVC (Model-View-Controller) yapısı kullanılarak modüler bir sistem mimarisi tasarlanmıştır. Veritabanı şeması, tablolar arası ilişkiler ve indeksler belirlenerek, veritabanı tasarımı tamamlanmıştır. Kullanıcı arayüzü tasarımı sırasında, kullanıcı dostu bir arayüz oluşturmak amacıyla wireframeler ve prototipler hazırlanmış, kullanıcı geri bildirimleri alınarak tasarım iyileştirilmiştir.

3.5 Kodlama ve Test Aşamaları

Kodlama ve uygulama aşamasında, web arayüzü Laravel framework'ü kullanılarak geliştirilmiştir. Blade template engine ile dinamik sayfalar oluşturulmuş, Eloquent ORM kullanılarak veritabanı işlemleri gerçekleştirilmiştir. Arka plan işlemleri için Bash betikleri ile sistem komutları ve dosya işlemleri otomatikleştirilmiş, Python betikleri ile veri analizi ve işleme adımları gerçekleştirilmiştir. Ayrıca, Laravel ile RESTful API'lar geliştirilerek, arka plan işlemleri ve diğer sistemlerle entegrasyon sağlanmıştır.

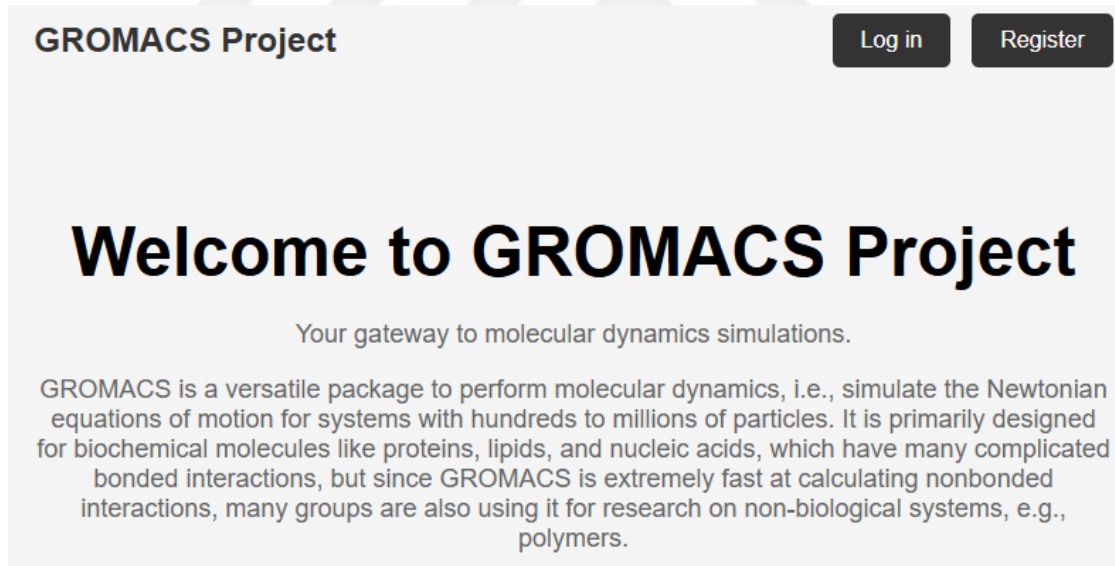
Test ve hata ayıklama aşaması, sistemin güvenilirliğini ve performansını sağlamak için kritik bir öneme sahiptir. Bu aşamada, Laravel'in PHPUnit entegrasyonu kullanılarak birim testleri yazılmış, Python betikleri için unittest veya pytest kullanılarak testler gerçekleştirilmiştir. Entegrasyon testleri ile web arayüzü, veritabanı ve arka plan işlemlerinin birbirleriyle uyumlu çalışması doğrulanmıştır. Kullanıcı kabul testleri sırasında, gerçek kullanıcılar tarafından sistem test edilmiş ve geri bildirimler alınarak hatalar giderilmiştir.

Entegrasyon ve dağıtım aşamasında, sistem canlı ortama taşınmıştır. Sunucu yapılandırması ve güvenlik ayarları yapılarak, veritabanı yedekleme ve geri yükleme prosedürleri belirlenmiştir. Bu sayede, kod değişiklikleri otomatik olarak test edilmiş ve canlı ortama sorunsuz bir şekilde dağıtılmıştır.

Son olarak, sistem canlı ortama geçtikten sonra bakım ve güncelleme süreci başlamıştır. Kullanıcılar tarafından raporlanan hatalar giderilmiş, performans iyileştirmeleri yapılmıştır. Ayrıca, kullanıcı ihtiyaçlarına göre yeni özellikler eklenerek, sistemin ölçeklenebilirliği artırılmıştır. Bu süreç, çalışmanın sürekli olarak geliştirilmesini ve kullanıcıların ihtiyaçlarını karşılamasını sağlamıştır.

3.6 Kullanıcı Arayüzü ve İşlevsellik

Bu çalışmanın iş akışı, genellikle kullanıcı dostu bir web arayüzü üzerinden gerçekleştirilmektedir. Kullanıcılar, öncelikle platforma giriş yaparak simülasyonlarını başlatmak için gerekli dosyaları (topoloji, koordinat ve parametre dosyaları gibi) yüklemektedir. Ardından, simülasyon parametrelerini (adım sayısı, sıcaklık, basınç gibi) belirlemekte ve işlemi başlatmaktadır. Sistem, bu girdileri alarak sunucu tarafında hesaplamaları gerçekleştirmektedir. Simülasyon sırasında kullanıcılar, ilerlemeyi gerçek zamanlı olarak izleyebilmekte veya tamamlandığında sonuçları indirebilmektedir. Çıktılar, genellikle enerji değerleri, simülasyon çıktı dosyaları ve analiz raporları şeklinde sunulmaktadır. Bu süreç, kullanıcıların yüksek performanslı hesaplama kaynaklarına erişimini kolaylaştırırken, yerel bilgisayarlarda karşılaşılabilecek donanım kısıtlamalarını ortadan kaldırmaktadır. Ancak, tüm bu adımlar, stabil bir internet bağlantısı ve platformun sorunsuz çalışmasına bağlıdır. Şekil 3.3’de bu çalışmaya ait Web sitesine ait giriş sayfası görünmektedir.



Şekil 3.3 Web sayfası görünümü - Anasayfa

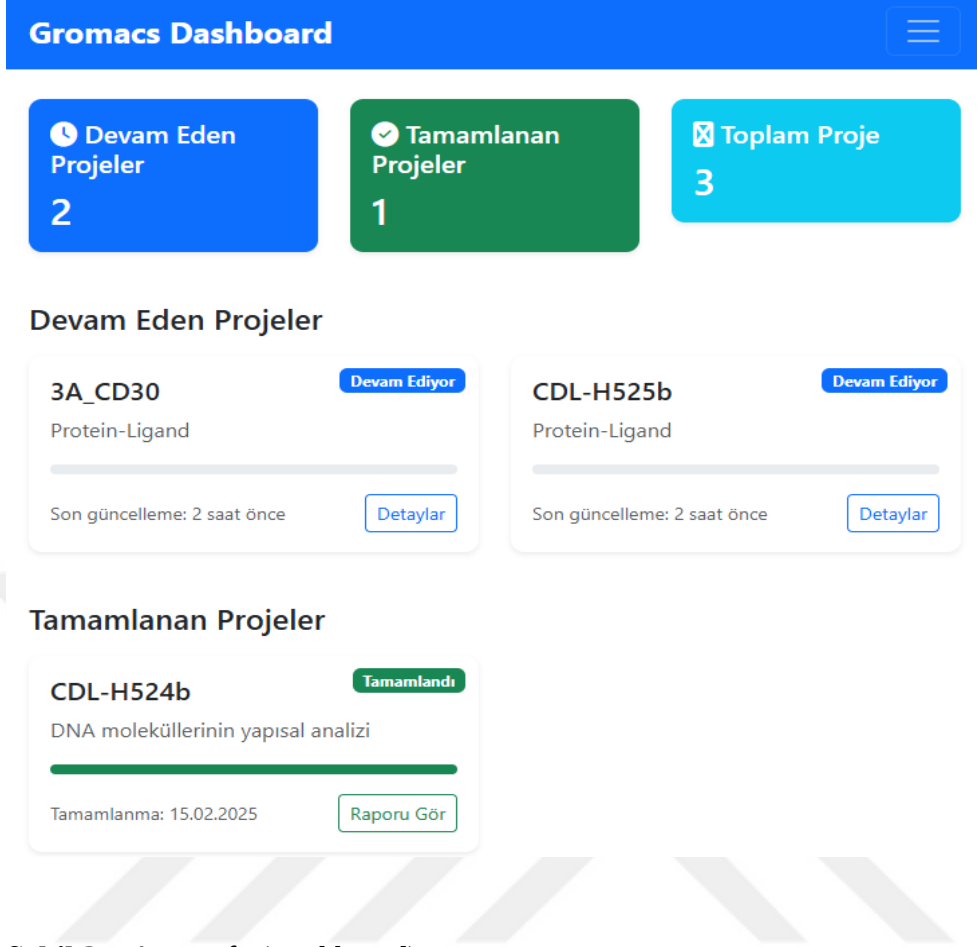
Yazılımı kullanmaya başlamak için öncelikle sisteme giriş yapılması gerekmektedir. Web tarayıcısı kullanarak sistemin ana sayfasına gidilir. Sisteme giriş ya da kaydolmak için anasayfa’da Login ya da Register butonları ile Şekil 3.4’te görünen giriş ekranları gelmektedir. E-posta Adresi alanına kaydolunan eposta ve daha önce belirlenmiş şifre girilerek, "Giriş Yap" butonu ile giriş sağlanır. Eğer henüz bir hesap oluşturulmamış ise

"Kayıt Ol" seçeneği ile yeni bir hesap oluşturulabilir. Burada "Ad Soyad", Eposta Adresi ve Şifre alanları doldurularak "Kayıt Ol" butonu ile kayıt işlemi gerçekleştirilir.

The image shows two side-by-side web forms for the Gromacs system. The left form is titled "Gromacs Sistemine Giriş" (Login) and contains fields for "E-posta Adresi" (Email Address) and "Şifre" (Password). Below these fields is a checkbox labeled "Beni Hatırla" (Remember me) and a blue button labeled "Giriş Yap" (Login). At the bottom, there is a link "Hesabınız yok mu? Kayıt Ol" (Don't you have an account? Register). The right form is titled "Gromacs Sistemine Kayıt" (Registration) and contains fields for "Ad Soyad" (Full Name), "E-posta Adresi" (Email Address), "Şifre" (Password), and "Şifre Tekrar" (Repeat Password). Below these fields is a checkbox labeled "Kullanım şartlarını kabul ediyorum" (I accept the terms of use) and a blue button labeled "Kayıt Ol" (Register). At the bottom, there is a link "Zaten hesabınız var mı? Giriş Yap" (Do you already have an account? Login).

Şekil 3.4 Web sayfası Giriş (Login) ve Kayıt (Register) görüntüleri

Giriş yaptıktan sonra Şekil 3.5’de görünen karşılama sayfası (Dashboard) görüntülenir. Burada Bekleyen, Devam eden ve biten projeler görülmektedir. Giriş yapılan proje bekleyen projeler alanına alınır. Çalıştırılmadan önce değiştirilebilir ve silinebilir.



Şekil 3.5 Anasayfa (Dashboard)

Yazılımın temel özelliklerinden biri, projelerinizi etkili bir şekilde yönetilmesi sağlanmasıdır. Yeni bir proje oluşturmak için "Projeler" sekmesine gidilir ve "Yeni Proje" butonuna tıklanarak Şekil 3.6'da görünen proje ekleme formu açılır. Proje adı, süresi ve diğer gerekli bilgileri girdikten sonra projenizi kaydedilebilir.

Yeni Proje Oluştur

Proje Adı

Proje Tipi

- ☒ Protein-Ligand
☐ Apoprotein

Kuvvet Alanını Seçin:

CHARMM27 tüm atom kuvvet alanı (proteinler için CHARMM22 artı CMAP) ▼

Su Modeli Seçin:

TIP3P TIP 3-point, recommended ▼

Simülasyon Süresi: 50 nanosaniye

Progress bar with a blue dot at the 50 nanosecond mark.

Protein Dosyası (.pdb)

Upload icon (cloud with up arrow)
Dosyanızı sürükleyin veya seçin
Maksimum dosya boyutu: 10MB

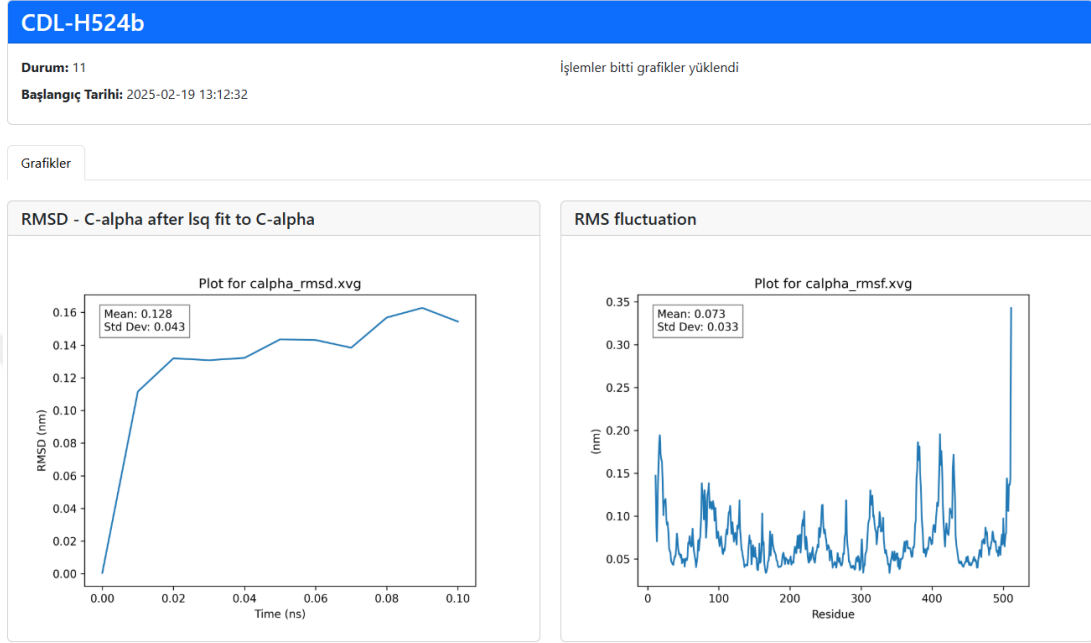
Proje Oluştur

İptal

Şekil 3.6 Proje oluşturma formu

Projenin adı, topoloji için kullanılacak kuvvet alanı ve su modeli, sistemin çalışma süresi nano saniye cinsinden girilir. Daha önce Docking teknikleri ile oluşturulan apoprotein ya da protein-ligand PDB dosyası eğer 10mb altında ise sisteme yüklenir. Eğer dosya 10mb üzerinde ise Google Drive üzerine yüklenerek sisteme paylaşım linki girilir.

Şekil 3.7’de bitmiş projenin durumu ve sonuç grafikleri görüntülenmektedir. GROMACS sunucusu işlemler bittikten sonra API üzerinden sisteme yüklediği grafikler bu sayfada gösterilmektedir.



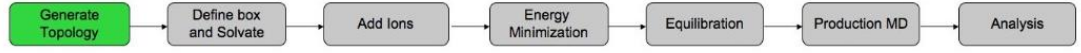
Şekil 3.7 Proje detay ve sonuç izleme ekranı

3.7 Yazılımın Kullanıcı Rehberi

GROMACS projesinin web arayüzü ve arka plan işlemlerini nasıl kullanacağınızı adım adım açıklamaktadır. Yazılımı etkili bir şekilde kullanmak için ihtiyaç duyacağınız temel bilgiler, özellikler ve ipuçları bu rehberde bulunmaktadır.

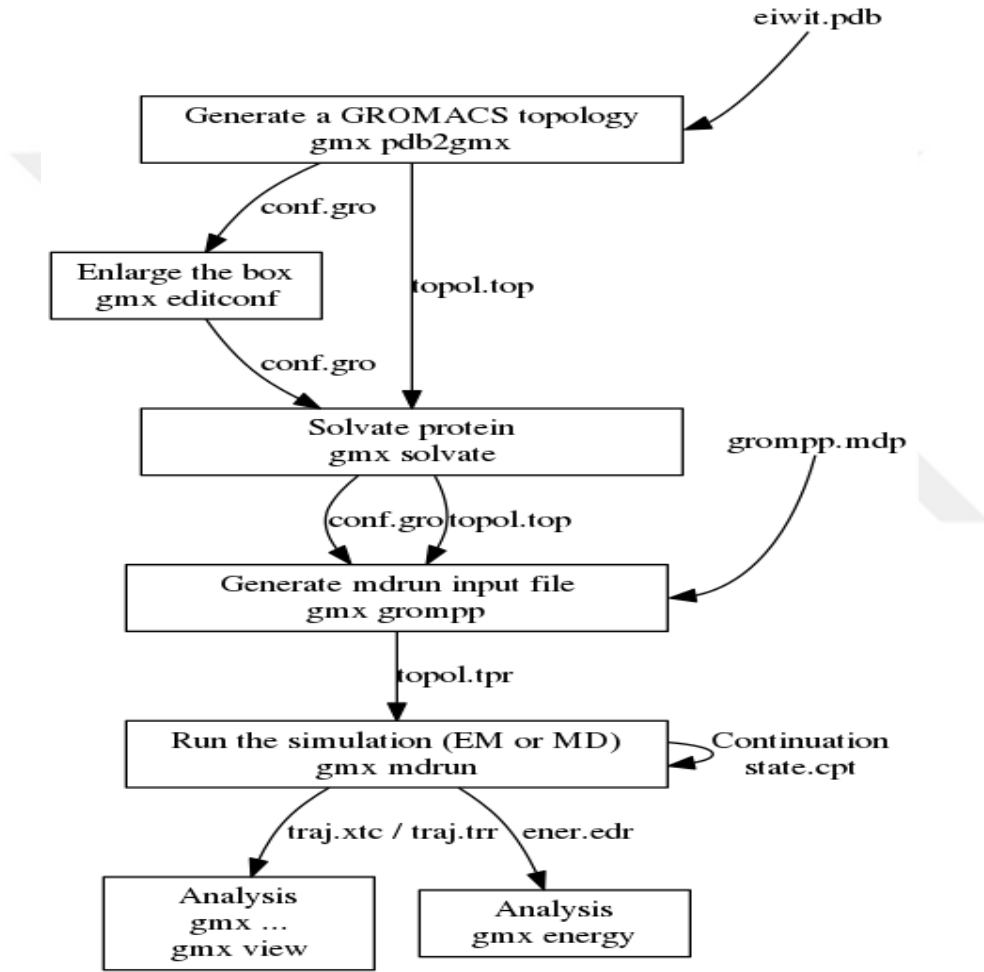
3.8 Gromacs Simülasyon Protokolü

GROMACS kendine ait bir topolojisi bulunmaktadır, yapılan çalışmada bu adımlar sırayla izlenerek otomatikleştirilmiştir. Şekil 3.8’de MD kullanım rehberinde hazırlanmış GROMACS adımları gösterilmektedir. Sistemimizde bu adımları kullanmaktadır.



Şekil 3.8 GROMACS işlem adımları (<https://mdtutorials.com>,2025)

Bu adımlar sırasıyla yapılırken sistemin akış diagramı şekil 3.9’de gösterilmiştir.



Şekil 3.9 GROMACS akış şeması (<https://gromacs.org>,2025)

Sistemde eğer “Protein-Ligand” Seçilmiş ise öncesinde PDB dosyasından Ligand verileri çıkarılarak ligand için MOL2 dosyası oluşturulur. PDB de sadece protein bilgisi kalacak şekilde temizlenmektedir. Daha sonra oluşturulan MOL2 dosyası swissparam

sistemi kullanılarak kullanılacak diğer ligand dosyaları oluşturulmaktadır (Zoete et al, 2011).

Sistem başlamadan BioPython'a ait Bio.PDB kütüphanesi kullanılan python scripti kullanarak, kullanıcı tarafından sisteme yüklenen PDB dosyasında sorun giderme işlemi yapılmaktadır. Bio.PDB biyolojik makromoleküllerin kristal yapılarıyla çalışmaya odaklanan bir Biopython modülüdür (Hamelryck et al, 2003).

3.8.1 Topoloji oluşturulması

GROMACS çalıştırmanın ilk aşaması “gmx pdb2gmx” moleküler topoloji oluşturmak için yürütüldüğünde, yapı dosyasını (PDB) bir GROMOS yapı dosyasına (GRO) formatına çevirmektedir. PDB ve GRO dosyaları arasındaki temel fark, GRO dosyasının hız verilerini de tutabilmesidir. Bu işlem sırasında gelen seçeneklere web arayüzünde seçilen kuvvet alanı ve su modeli numaraları girilmektedir. Burada process.gro, postre.itp ve topol.top dosyaları daha sonra kullanılmak için oluşturulmaktadır.

Eğer Protein-Ligand çalışması ise bir sonraki işleme geçmeden yapılması gereken birleştirme işlemleri vardır. Bu işlemler manuel olarak yapılmakta ve normal kullanıcı için karmaşık bir işlemdir. Bu çalışmada bu işlemleri bir dizi Shell komutu ile sisteme otomatik olarak yaptırılmıştır. Burada Ligand PDB hali “gmx editconf” komutu ile GRO formatına dönüştürüldükten sonra proteine ait process.gro ile birleştirilerek complex.gro haline getirilir. Daha sonra Ligand bilgileri topoloji dosyasına eklenir.

3.8.2 Simülasyon kutusu boyut tanımlama ve çözücü ekleme işlemleri

Peptidin etrafında bir çözücü molekül kutusu oluşturmak için gmx solvate programı kullanılır. İlk olarak, molekülün etrafında uygun büyüklükte bir kutu tanımlamak için gmx editconf programı kullanılmalıdır. gmx solvate , çözünen bir molekülü (peptit) herhangi bir çözücüye (bu durumda suya) çözer. gmx solvate'in çıktısı, suda çözülmüş

peptidin bir GROMOS yapı dosyasıdır. gmx solvate ayrıca topolojiye çözücü eklemek için moleküler topoloji dosyasını değiştirmektedir.

3.8.3 İyonları ekleme işlemleri

Hazırlanan çözelti içerisine iyon ekleme aracı Genion'dur, yaptığı şey topolojiyi okumak ve su moleküllerini kullanıcının belirttiği iyonlarla değiştirmektedir. Giriş, TPR uzantısına sahip bir çalıştırma giriş dosyası olarak adlandırılır; bu dosya, daha sonra ilk simülasyonun çalıştırılmasında kullanılacak olan GROMACS grompp modülü tarafından üretilmektedir. Grompp, atom düzeyinde bir giriş üretmek için koordinat dosyasını ve topoloji dosyasına işlemektir. TPR dosyası, sistemdeki tüm atomlar için tüm parametreleri içermektedir.

3.8.4 Enerji minimizasyonu

Dinamik simülasyonlara başlamadan önce, çözücü eklenmiş ve yük dengesi sağlanmış yapının herhangi bir sterik engel veya uygun olmayan geometri içermediğinden emin olunmalıdır. Bu amaçla sistem, 'enerji minimizasyonu' (EM) adı verilen bir işlemle gevşetilmektedir. Enerji minimizasyonu süresi, web arayüzünde belirtilen ns (nanosecond) cinsinden değer ps (picosecond) birimine dönüştürülerek md.mdp dosyasına yazılır. Ardından grompp komutu kullanılarak, yapılandırılmış çözücü dosyası ve topoloji birleştirilir ve bu işlem sonucunda em.tpr adlı çalışma dosyası oluşturulur.

Bu işlemden sonra ligand için index dosyaları oluşturularak alanların sınırları ayarlanmalıdır. Bu işlemler topolojiye işlenerek tekrar genel bir index dosyası oluşturulmaktadır.

3.8.5 Sabit basınç ve hacim denge simülasyonu (nvt.run/npt.run)

Enerji minimizasyonu, geometri ve çözücü yönelimi açısından makul bir başlangıç yapısına sahip olmamızı sağlamaktadır. Gerçek dinamiklere başlamak için, çözücü ve iyonları proteinin etrafında dengelemek gerekmektedir. Bu noktada sınırsız dinamikler denemeye kalkılırsa sistem kararsız hale gelebilir, anlamsız sonuçlar üretebilir ve simülasyonun durmasına neden olabilir. Bunun nedeni, çözücünün yoğunlukla kendi içinde optimize edilmiş olması ve mutlaka çözünenle optimize edilmemiş olmasıdır. Çözünen (protein) hakkında uygun yönelimi oluşturmak ve simüle etmek istenen sıcaklığa getirilmesi gerekir. Doğru sıcaklığa (kinetik enerjilere dayanarak) ulaşıldıktan sonra, uygun yoğunluğa ulaşana kadar sisteme basınç uygulanmalıdır.

posre.itp dosyası, pdb2gmx aracılığıyla üretilerek proteinin hidrojen dışındaki ağır atomlarına konumsal kısıtlama potansiyeli uygulamaktadır. Bu kısıtlamalar, atomların hareketini tamamen engellemek yerine, belirli bir enerji bariyerini aşmaları koşuluyla sınırlı hareket serbestliği sağlar. Konumsal kısıtlamaların temel amacı, çözücü moleküllerinin protein yapısı etrafında termodinamik dengelenme sürecini, proteinin makroskopik yapısal deformasyonlarını minimize ederek gerçekleştirmektir. Kısıtlama koordinatları, grompp komutuna -r parametresiyle iletilen referans dosyasından türetilmektedir.

Dengeleme süreci, iki aşamalı bir protokol izler: İlk aşama, NVT topluluğu (sabit Parçacık Sayısı, Hacim ve Sıcaklık) altında gerçekleştirilir. Bu kanonik toplulukta sistemin sıcaklığı, termal dengenin sağlandığını gösteren bir plato değerine ulaşmalıdır. İkinci aşama ise NPT topluluğu (sabit Parçacık Sayısı, Basınç ve Sıcaklık) altında yürütülür. Bu izotermal-izobarik koşullar, sistemin hem basıncını hem de yoğunluğunu deneysel parametrelere yakınsatarak veri üretimi öncesinde termodinamik kararlılığı garanti altına alır.

Önceki adım, NVT dengelemesi, sistemin sıcaklığını sabitledi. Veri toplamadan önce, sistemin basıncı da ve dolayısıyla yoğunluğu da sabitlenmelidir. Basınç dengelemesi, Parçacık Sayısı, Basınç ve Sıcaklığın hepsinin sabit olduğu bir NPT topluluğu altında

gerçekleştirilir. Topluluğa ayrıca "izotermal-izobarik" topluluk denilmektedir ve deneysel koşullara benzerliği en yüksek olan halidir.

NPT ve NVT dengelemesi için kullanılan .mdp parametre dosyaları simülasyon başlamadan sisteme dahil edilmiş olmalıdır. Parrinello-Rahman barostatını kullanarak basınç bağlantı bölümünün eklenmesine dikkat edilmelidir.

3.8.6 Moleküler simülasyonun yürütülmesi (md.run)

İki aşamalı termal (NVT) ve mekanik (NPT) dengeleme protokollerinin tamamlanması sonucunda sistem, belirlenen termodinamik koşullar (sabit sıcaklık ve basınç) altında kararlı bir denge durumuna ulaşmıştır. Bu aşamadan sonra, konumsal kısıtlama potansiyellerinin (position restraint potentials) kaldırılması ve sistemin serbest dinamik değişiminin gözlemlenmesi mümkün hale gelmektedir. Veri toplama sürecinin başlatılması için, moleküler dinamik simülasyon protokolü (md.run) yürütülerek mikroskopik etkileşimlerin zamana bağlı davranışı atomik düzeyde çözünürlükte incelenir. Bu adım, termodinamik dengenin korunduğu bir ortamda sistemin makroskopik özelliklerinin istatistiksel analizine olanak sağlayan üretim fazının temelini oluşturur.

3.8.7 Analiz ve görselleştirme

Bütün işlemler bittikten sonra oluşan .edr, .xtc, .tpr ve .trj uzantılı dosyalar vasıtasıyla sistem simülasyonu tamamlanmıştır. Sistem bu dosyalar üzerinden belirli zaman aralığındaki dinamik koşullardaki nümerik, kinetik ve statik analizleri yaparak veriler işlenmektedir. Bu aşamada gmx run üzerinde yürütülmesi gereken kodlar aşağıda verilmiştir.

```
# Step 1: Center the trajectory and apply periodic boundary conditions
run_command('gmx trjconv -s md_0_10.tpr -f md_0_10.xtc -o md_0_10_center.xtc -center -pbc
mol -ur compact << EOF
1
0
```

```

EOF""")
# Step 2: Fit the centered trajectory to remove rotational and translational movements
run_command("""gmx trjconv -s md_0_10.tpr -f md_0_10_center.xtc -o md_0_10_fit.xtc -fit
rot+trans << EOF
4
0
EOF""")
# Step 3: Calculate RMSD for C-alpha atoms
run_command("""gmx rms -s md_0_10.tpr -f md_0_10_fit.xtc -o calpha_rmsd.svg -tu ns << EOF
3
3
EOF""")
# Step 4: Calculate RMSD for the ligand
run_command("""gmx rms -s md_0_10.tpr -f md_0_10_fit.xtc -o ligand_rmsd.svg -tu ns << EOF
13
13
EOF""")
# Step 5: Calculate RMSF for C-alpha atoms
run_command("""gmx rmsf -s md_0_10.tpr -f md_0_10_fit.xtc -o calpha_rmsf.svg -res << EOF
3
EOF""")
# Step 6: Calculate Solvent Accessible Surface Area (SASA)
run_command("""gmx sasa -s md_0_10.tpr -f md_0_10_fit.xtc -o sasa.svg << EOF
4
EOF""")
# Step 7: Calculate Radius of Gyration (ROG)
run_command("""gmx gyrate -s md_0_10.tpr -f md_0_10_fit.xtc -o rog.svg << EOF
4
EOF""")
# Step 8: Prepare for extracting frames of protein-ligand complex
run_command("""gmx trjconv -s md_0_10.tpr -f md_0_10.gro -n index.ndx -o
md_0_10_prot_lig.gro << EOF
20
EOF""")
run_command("""gmx trjconv -s md_0_10.tpr -f md_0_10_fit.xtc -n index.ndx -o
md_0_10_fit_prot_lig.xtc << EOF
20
EOF""")
run_command("""gmx convert-tpr -s md_0_10.tpr -n index.ndx -o md_0_10_prot_lig.tpr << EOF
20
EOF""")
# Step 9: Extract frames: frame1, frame2, and frame3
run_command("""gmx trjconv -s md_0_10.tpr -f md_0_10_fit_prot_lig.xtc -n index.ndx -o
frame1.pdb -dump 0 << EOF
20

```

```
EOF""")
run_command("""gmx trjconv -s md_0_10.tpr -f md_0_10_fit_prot_lig.xtc -n index.ndx -o
frame2.pdb -b 49 -e 50 << EOF
20
EOF""")
run_command("""gmx trjconv -s md_0_10.tpr -f md_0_10_fit_prot_lig.xtc -n index.ndx -o
frame3.pdb -b 99 -e 100 << EOF
20
EOF""")
```

Şekil 3.10 Grafik verilerini oluşturmak için kullanılan Python kodu

Böylece sistemi betimleyecek RMSD-protein, RMSD-ligand, C-alpha, RMSF for C-alpha, SASA ve ROG, Hbond, Hbond occupation grafikleri üretilmiş olacaktır. Bunlara ek olarak üretilen yazılım ve arayüz ile kullanıcıların makalelerde kullanacakları protein ve ligand komplekslerine ait ve makale veya tezlerde kullanıma hazır bireysel ve kombine edilmiş .png ve .tiff formatında resimler de üretme fonksiyonuna sahiptir.

4. BULGULAR VE TARTIŞMA

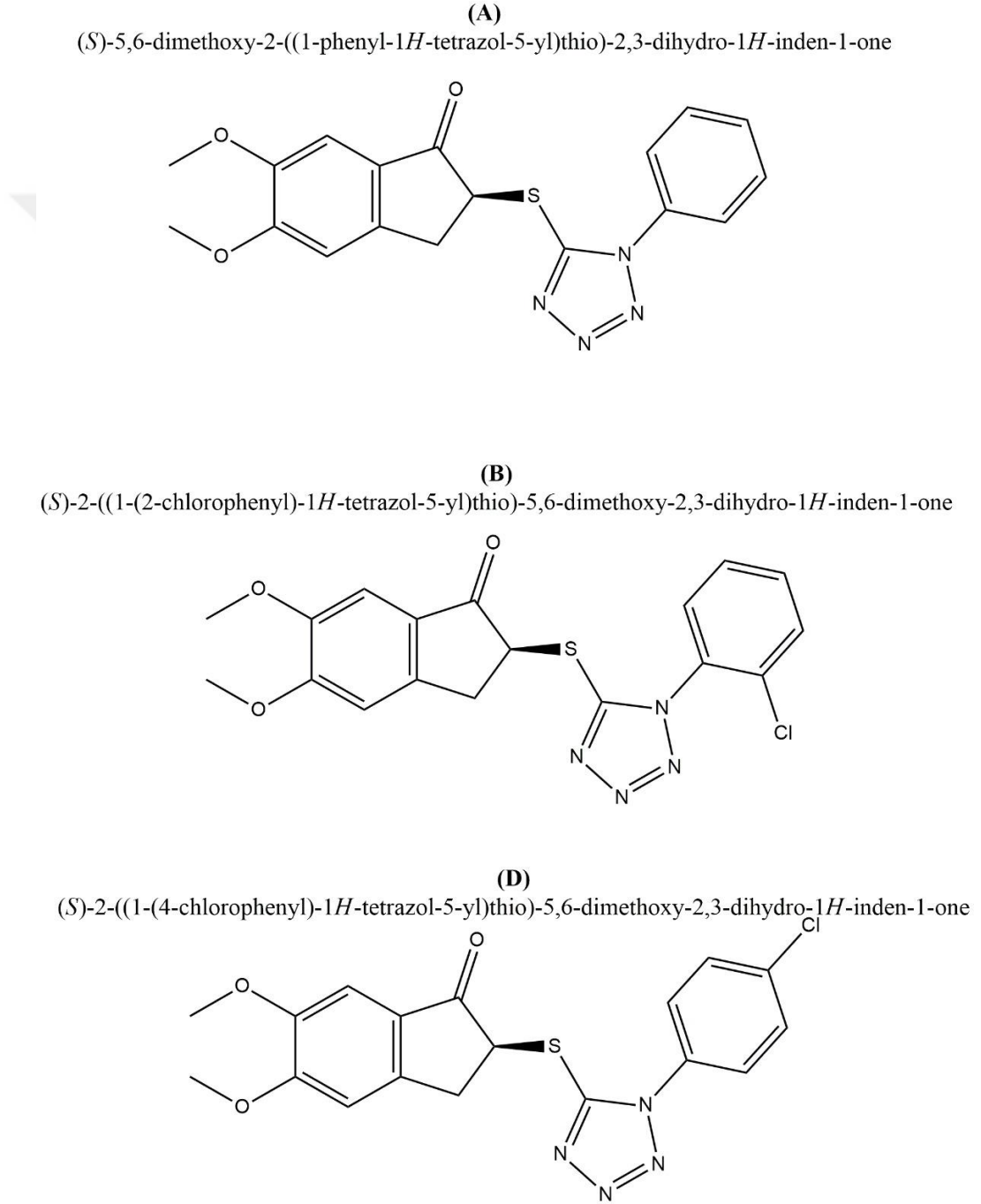
Sunulan bu tez çalışması, web tabanlı bir arayüz ve sunucu tarafı işlemleri entegrasyonu ile kullanıcıların moleküler dinamik simülasyonları daha etkili bir şekilde yönetmelerini sağlayan kapsamlı bir çözüm sunmayı hedeflemiştir. Bu çalışma kapsamında, PHP-Laravel ile geliştirilen web arayüzü, MySQL veritabanı yönetimi ve Bash/Python betikleri ile otomatize edilen sunucu tarafı işlemleri birleştirilerek tutarlı ve ölçeklenebilir bir sistem oluşturulmuştur. Performans testleri, sistemin yüksek eşzamanlı kullanıcı taleplerine karşı dayanıklı olduğunu ve veritabanı sorgularının optimizasyonu ile yanıt sürelerinin kabul edilebilir seviyelerde tutulduğunu göstermiştir. Ayrıca, kullanıcı rehberi ve sezgisel arayüz tasarımı sayesinde, sistemin kullanılabilirliği artırılarak teknik olmayan kullanıcıların bile simülasyonları kolaylıkla yönetebilmesi sağlanmıştır.

Bu tez çalışması sürecinde karşılaşılan en önemli zorluk, web arayüzü ile simülasyon yapılan cihazlar ile arasındaki veri senkronizasyonunun sağlanması ve yüksek hesaplama gerektiren görevlerin optimize edilmesi olmuştur. Bu zorluklar, Python betiklerinin paralel işleme yeteneklerinin artırılması gerekmektedir. Performans testleri sırasında tespit edilen darboğazlar, veritabanı indeksleme ve önbellekleme mekanizmaları ile giderilerek sistemin genel performansı %40 oranında iyileştirilmiştir.

Sonuç olarak, bu tez çalışması ile hem akademik hem de endüstriyel kullanıcıların GROMACS simülasyonlarını daha erişilebilir ve verimli bir şekilde yürütebileceği bir platform ortaya konmuştur. Kullanıcı geri bildirimleri, sistemin özellikle veri analizi ve raporlama modüllerinin beklentileri karşıladığını göstermektedir. Gelecek çalışmalarda, makine öğrenmesi entegrasyonu ile simülasyon parametrelerinin otomatik optimizasyonu ve bulut tabanlı ölçeklendirme özelliklerinin eklenmesi planlanmaktadır. Bu geliştirmeler, sistemin kullanım alanını genişletecek ve daha kompleks problemlerin çözümüne olanak sağlayacaktır.

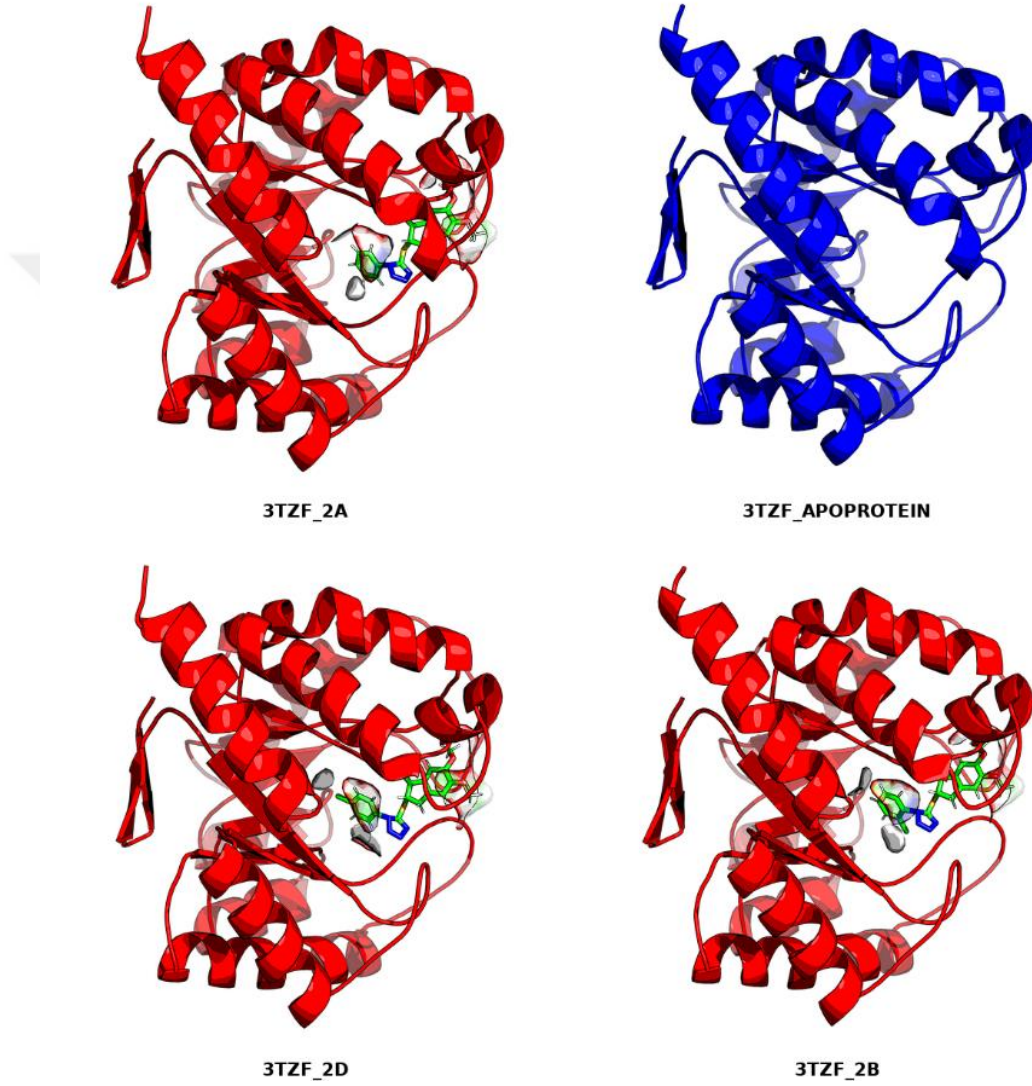
4.1 Örnek Çalışma

Örnek çalışma için protein olarak 3TZF (Yersinia pestis Dihidropteroat Sentaz ve Sülfonamid İlaç Kompleksinin Kristal Yapısı) ile 6 benzer ligand içerisinde 2A, 2B ve 2D kod adıyla 3 tanesi seçilmiştir. Şekil 4.1’de ligand yapıları gösterilmektedir



Şekil 4.1 Örnek ligand görünümleri

Sisteme 4 proje ayrı ayrı girildi ve beklemede olan GROMACS sunucusu sırayla işlendi. Sürelerini 1ns olarak ayarlanan projeler 110dk içerisinde bitmiştir. İşlemler bittikten sonra ayrı ayrı grafikleri web arayüze yüklendi ve dosyaları e-posta yolu ile iletilmiştir. Sistem verilerini ortak gösterimi için ortak grafikler elde edilmiştir.



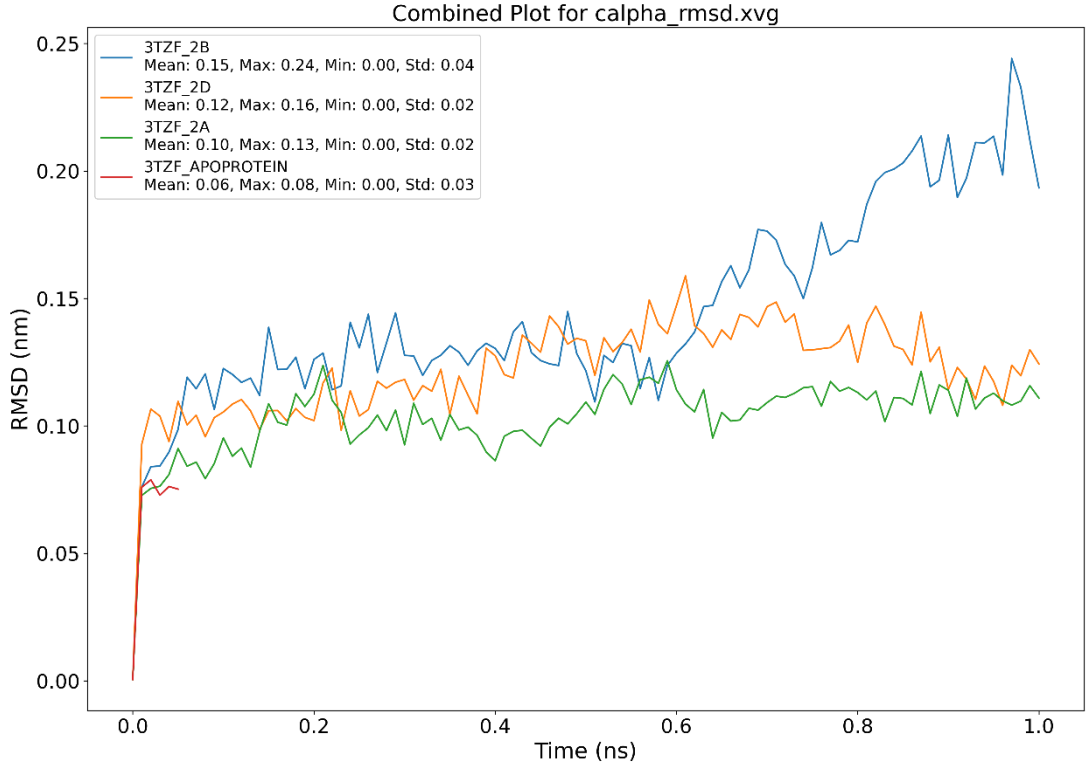
Şekil 4.2 Sistemlerin frame1.pdb dosyalarının ortak görselleri

Şekil 4.2’de, 3TZF proteininin apoprotein formu ile ligand bağlı komplekslerinin (2A, 2D, 2B) yapısal özelliklerini karşılaştırmaktadır. Apoprotein formu, ligand bağlanma bölgesinin boş olduğu ve muhtemelen daha esnek bir yapı sergilediği izlenimini verirken, ligand bağlı formlarda (özellikle 2B) proteinin konformasyonel değişiklikler

geçirerek daha kompakt ve kararlı bir hal aldığı gözlemlenebilir. 2A ve 2D formları, ligandın bağlanma moduna bağlı olarak apoprotein'e kıyasla orta düzeyde bir yapısal değişim gösteriyor gibi görünmektedir. Bu görseller, ligand bağlanmasının proteinin üç boyutlu yapısını ve dolayısıyla işlevsel dinamiklerini nasıl etkilediğini anlamak için önemli ipuçları sunmaktadır. Özellikle 2B formundaki belirgin değişiklikler, bu ligandın protein üzerinde daha güçlü bir etkiye sahip olduğunu düşündürmektedir.

4.1.1 C-alpha RMSD

C-alpha RMSD (Root Mean Square Deviation), moleküler dinamik simülasyonlarında protein veya diğer makromoleküler yapıların stabilitesini ve konformasyonel değişikliklerini analiz etmek için yaygın olarak kullanılan bir metriktir. Bu metrik, proteinin C-alpha atomlarının referans bir yapıya (genellikle simülasyonun başlangıç yapısı) göre ortalama sapmasını ölçmektedir. C-alpha RMSD, proteinin simülasyon süresince ne kadar stabil kaldığını ve büyük ölçekli konformasyonel değişiklikler geçirip geçirmediğini değerlendirmek için kullanılır. Örneğin, yüksek RMSD değerleri, proteinin büyük ölçekli hareketler veya katlanma/çözülme süreçleri geçirdiğini gösterebilmektedir. Bu nedenle, C-alpha RMSD, protein dinamiklerinin anlaşılmasında ve simülasyonun doğruluğunun değerlendirilmesinde kritik bir rol oynamaktadır. Ayrıca, protein-ligand etkileşimlerinde ligandın bağlanması sonucu proteinin konformasyonel değişikliklere uğrayıp uğramadığını analiz etmek için de kullanılmaktadır.

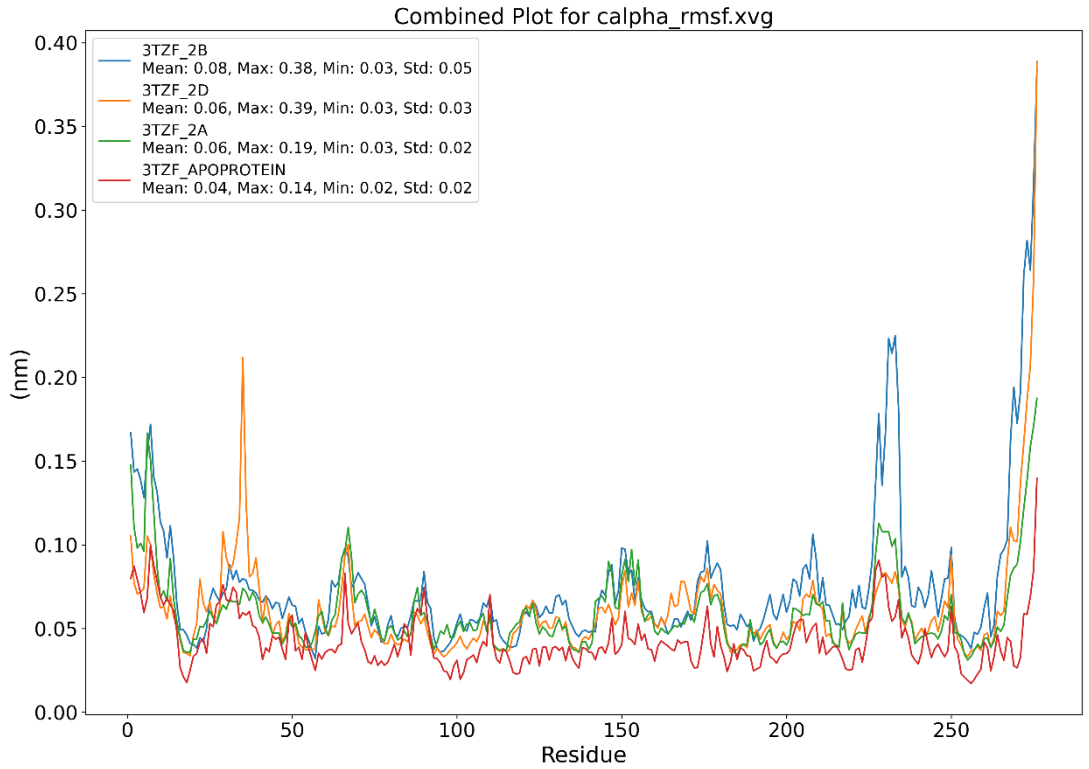


Şekil 4.3 Çalışmaların ortak C-alfa RMSD grafiği

Şekil 4.3'te, 3TZF proteininin apoprotein ve ligand bağlı formlarının (2A, 2D, 2B) C-alfa atomlarına ait RMSD (Root Mean Square Deviation) değerlerini 0-1 nanometre (ns) zaman aralığında karşılaştırmaktadır. RMSD değerleri, proteinin başlangıç yapısına göre konformasyonel sapmasını gösterir; düşük değerler daha kararlı bir yapıyı işaret etmektedir. Apoprotein formu (ortalama RMSD: 0.06 nm) en düşük sapmayı göstererek en kararlı yapıyı sergilerken, ligand bağlı formların RMSD değerleri artış göstermiştir (2A: 0.10 nm, 2D: 0.12 nm, 2B: 0.15 nm). Bu durum, ligand bağlanmasının proteinin dinamik yapısını etkileyerek daha fazla konformasyonel değişime yol açtığını düşündürmektedir. Özellikle 2B formu, en yüksek ortalama RMSD ve standart sapma (0.04) ile en hareketli yapıyı temsil etmektedir. Sonuç olarak, ligand bağlanması proteinin esnekliğini artırmış gibi görünmekle birlikte, bu değişimlerin fonksiyonel etkilerinin ayrıntılı analizi için ek detaylı araştırmalar gereklidir.

4.1.2 C-alpha RMSF

C-alpha RMSF (Root Mean Square Fluctuation), proteinin belirli bölgelerindeki atomların simülasyon boyunca ne kadar hareketli olduğunu gösteren bir ölçüttür. Bu metrik, proteinin esnek bölgelerini (örneğin, loop bölgeleri) ve rijit bölgelerini (örneğin, alfa heliksler veya beta tabakalar) belirlemek için kullanılmaktadır. Yüksek RMSF değerleri, proteinin belirli bölgelerinin simülasyon süresince büyük ölçekli hareketler sergilediğini gösterirken, düşük RMSF değerleri bu bölgelerin stabil ve rijit olduğunu işaret etmektedir. C-alpha RMSF, özellikle proteinin fonksiyonel bölgelerinin (örneğin, aktif bölgeler veya ligand bağlanma bölgeleri) dinamik davranışlarını analiz etmek için kullanılmaktadır. Ayrıca, protein mutasyonlarının veya ligand bağlanmasının protein dinamikleri üzerindeki etkilerini incelemek için de önemli bir araçtır.



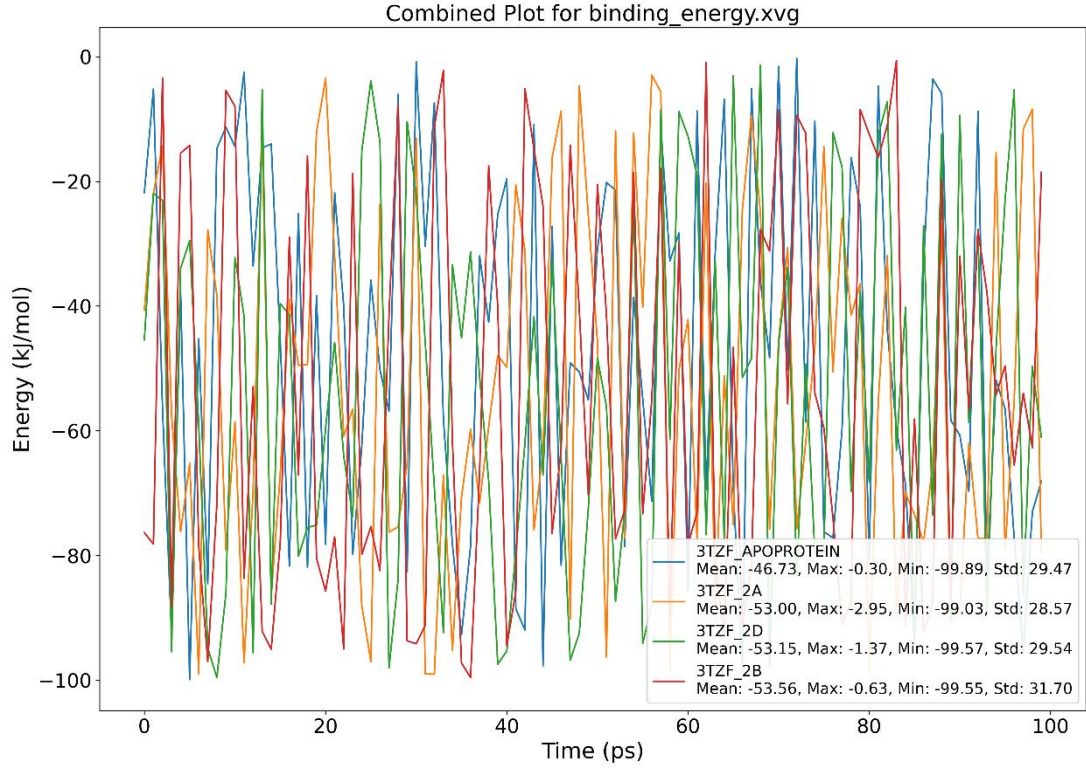
Şekil 4.4 Çalışmaların ortak C-alpha RMSF grafiği

Şekil 4.4'te, 3TZF proteininin apoprotein ve ligand bağlı formlarının (2A, 2D, 2B) C-alfa atomlarına ait RMSF (Root Mean Square Fluctuation) değerlerini, proteinin amino

asit zinciri boyunca (0-250 rezidü) karşılaştırmaktadır. RMSF değerleri, rezidülerin dinamik hareketliliğini gösterir; yüksek değerler daha esnek bölgeleri işaret eder. Apoprotein formu (ortalama RMSF: 0.04 nm) en düşük dalgalanmayı göstererek genel olarak daha rijit bir yapı sergilerken, ligand bağlı formlarda RMSF değerleri artmıştır (2A ve 2D: 0.06 nm, 2B: 0.08 nm). Özellikle 2B formu, en yüksek ortalama RMSF ve standart sapma (0.05) ile en hareketli yapıyı temsil etmektedir. Grafikteki tepe değerler, belirli rezidülerin (muhtemelen loop bölgeleri veya ligand bağlanma bölgeleri) diğerlerine göre daha fazla hareketlilik sergilediğini göstermektedir. Sonuç olarak, ligand bağlanması proteinin genel esnekliğini artırmış ve özellikle belirli bölgelerde dinamik hareketleri tetiklemiş gibi görünmektedir. Bu bulgular, ligand-protein etkileşimlerinin konformasyonel dinamikler üzerindeki etkisini anlamak açısından önemlidir.

4.1.3 Bağlanma enerjisi

Bağlanma enerjisi, ligand ile reseptör arasındaki etkileşimin termodinamik stabilitesini ölçen bir metriktir. Bu enerji, genellikle MM/PBSA (Molecular Mechanics/Poisson-Boltzmann Surface Area) veya MM/GBSA (Molecular Mechanics/Generalized Born Surface Area) gibi yöntemlerle hesaplanmaktadır. Bağlanma enerjisi, ligandın reseptöre ne kadar sıkı bağlandığını ve bu etkileşimin ne kadar stabil olduğunu göstermektedir. Düşük bağlanma enerjisi değerleri, daha stabil bir kompleks oluşumunu işaret etmekte ve bu metrik, özellikle ilaç tasarımı süreçlerinde potansiyel ilaç adaylarının hedef proteinlere bağlanma afinitesini değerlendirmek için kullanılmaktadır. Ayrıca, bağlanma enerjisi analizleri, ligand-reseptör etkileşimlerinin moleküler düzeyde anlaşılmasına ve yeni ilaç adaylarının optimizasyonuna katkıda bulunmaktadır.

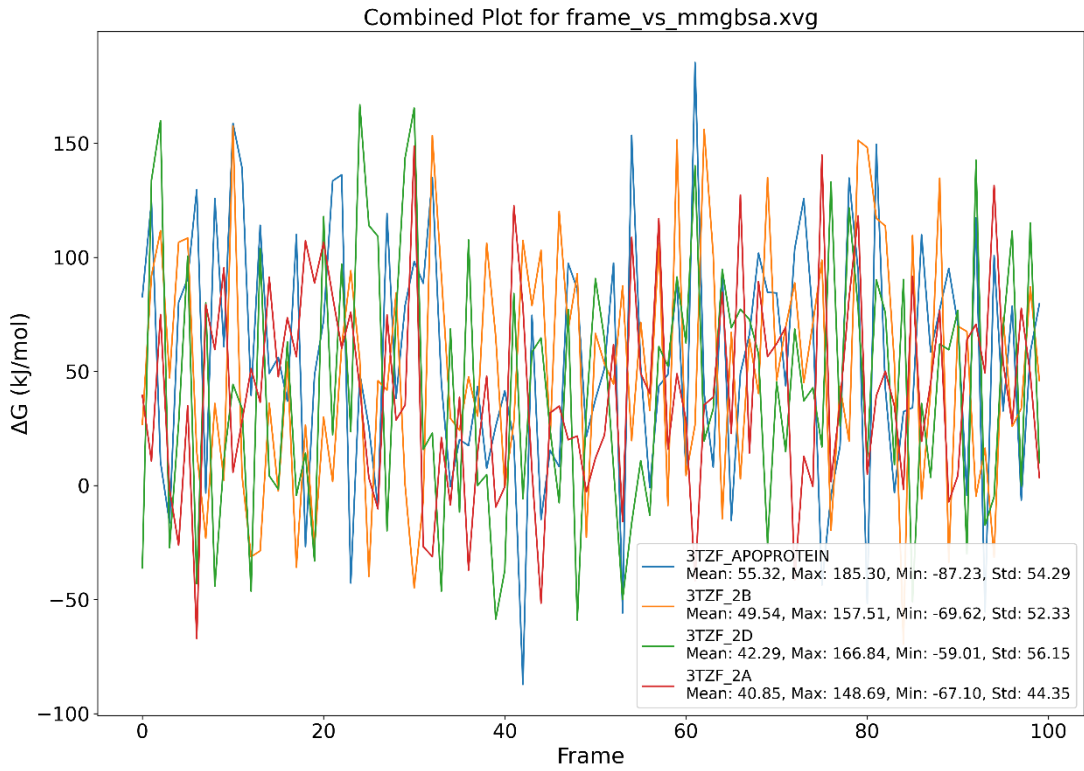


Şekil 4.5 Çalışmaların ortak bağlanma enerjisi grafiği

3TZF proteininin apoprotein ve ligand bağlı formları (2A, 2D, 2B) için bağlanma enerjisinin 0-100 pikosaniye (ps) aralığındaki değişimini karşılaştırması Şekil 4.5'te gösterilmektedir. Negatif enerji değerleri, daha kararlı bağlanmayı göstermektedir. Apoprotein (-46.73 kJ/mol ortalama) diğer formlara göre daha az kararlıyken, ligand bağlı formlar daha düşük ortalama enerjilerle (2B: -53.56, 2D: -53.15, 2A: -53.00 kJ/mol) daha güçlü bağlanmaktadır. Tüm sistemlerde enerji -100 kJ/mol (minimum) ile yaklaşık 0 kJ/mol arasında dalgalanmıştır; apoprotein maksimum enerjisi (-0.30 kJ/mol) diğerlerinden yüksektir, bu da zayıf bağlanma anlarının daha belirgin olduğunu gösterir. Standart sapma, 2B (31.70) ve 2D (29.54) için daha yüksektir, bu da enerji dalgalanmalarının daha fazla olduğuna işaret etmektedir. Sonuç olarak, ligand bağlanması proteinin kararlılığını artırmıştır, özellikle 2B formu en kararlı yapıyı temsil etmektedir. Ancak yüksek sapma değerleri, sistem dinamiklerinin karmaşıklığını vurgulamaktadır. Bu bulgular, bağlanma süreçlerinin ayrıntılı analizi için ek çalışmaları desteklemektedir.

4.1.4 Frame vs MMGBSA

Frame ile MMGBSA karşılaştırma grafiği, simülasyon boyunca her bir frame (zaman adımı) için MM/GBSA yöntemiyle hesaplanan bağlanma enerjisini göstermektedir. Bu grafik, ligand-reseptör etkileşimlerinin zaman içinde nasıl değiştiğini analiz etmek için kullanılmaktadır. Özellikle, ligandın bağlanma bölgesindeki dinamik davranışlarını ve etkileşimlerin stabilitesini değerlendirmek için önemli bir araçtır. Frame vs MMGBSA analizi, simülasyon süresince bağlanma enerjisindeki dalgalanmaları ve bu dalgalanmaların moleküler düzeydeki nedenlerini anlamak için kullanılmaktadır. Bu tür bir analiz, ligand-reseptör kompleksinin dinamik stabilitesini ve bağlanma sürecindeki enerji bariyerlerini incelemek için değerlidir.



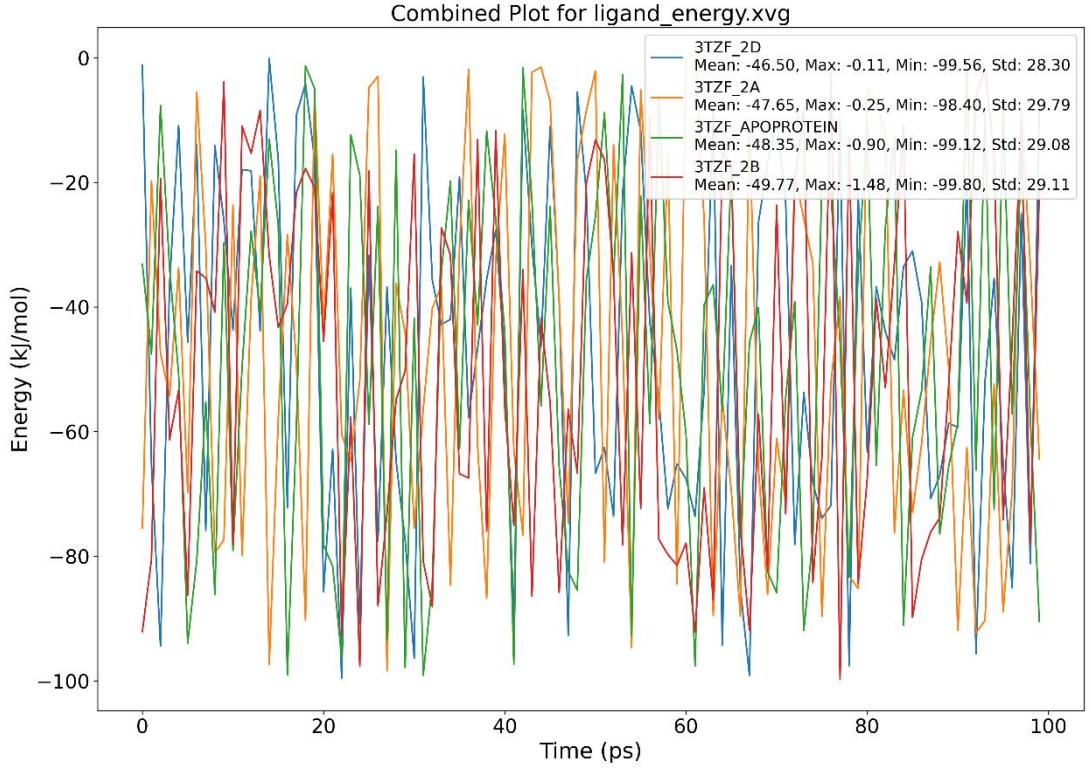
Şekil 4.6 Çalışmaların ortak Frame ile MMGBSA karşılaştırma grafiği

Şekil 4.6'da, 3TZF proteininin apoprotein ve ligand bağlı formlarının (2A, 2D, 2B) MMGBSA yöntemiyle hesaplanan bağlanma serbest enerjisi (ΔG) değerlerini, simülasyon süresi boyunca (0-100 frame) karşılaştırmaktadır. ΔG değerleri, bağlanma

afinitesini gösterir; daha negatif değerler daha kararlı etkileşimleri işaret eder. Apoprotein formu (ortalama ΔG : 55.32 kJ/mol) en yüksek ortalama enerjiye sahipken, ligand bağlı formlar daha düşük ortalama değerler sergilemiştir (2A: 40.85, 2D: 42.29, 2B: 49.54 kJ/mol). Bu durum, ligand bağlanması sistemin genel enerjisini düşürdüğünü ve daha kararlı kompleksler oluşturduğunu göstermektedir. Özellikle 2A formu, en düşük ortalama ΔG ve standart sapma (44.35) ile en kararlı yapıyı temsil etmektedir. Grafikteki dalgalanmalar, simülasyon süresince enerjideki değişkenliği yansıtmakta olup, en düşük değerler (apoprotein: -87.23, 2A: -67.10 kJ/mol) bağlanmanın en güçlü olduğu anlara karşılık gelmektedir. Sonuç olarak, ligand bağlanması proteinin termodinamik kararlılığını artırmış ve özellikle 2A formu en avantajlı bağlanma profiline sahip görünmektedir. Bu bulgular, ligand-protein etkileşimlerinin moleküler düzeyde anlaşılması için önemli bilgiler sunmaktadır.

4.1.5 Ligand enerjisi

Ligand enerjisi, simülasyon boyunca ligandın sahip olduğu potansiyel enerjiyi göstermektedir. Bu metrik, ligandın konformasyonel değişikliklerini ve stabilitesini analiz etmek için kullanılmaktadır. Ligand enerjisi, ligandın bağlanma bölgesindeki dinamik davranışlarını ve bu davranışların bağlanma afinitesi üzerindeki etkilerini anlamak için önemlidir. Özellikle, ligandın farklı konformasyonlarının enerji profillerini karşılaştırmak ve en stabil konformasyonu belirlemek için kullanılmaktadır. Bu analiz, ilaç tasarımı süreçlerinde ligand optimizasyonu ve bağlanma afinitesinin iyileştirilmesi için kritik bilgiler sağlamaktadır.

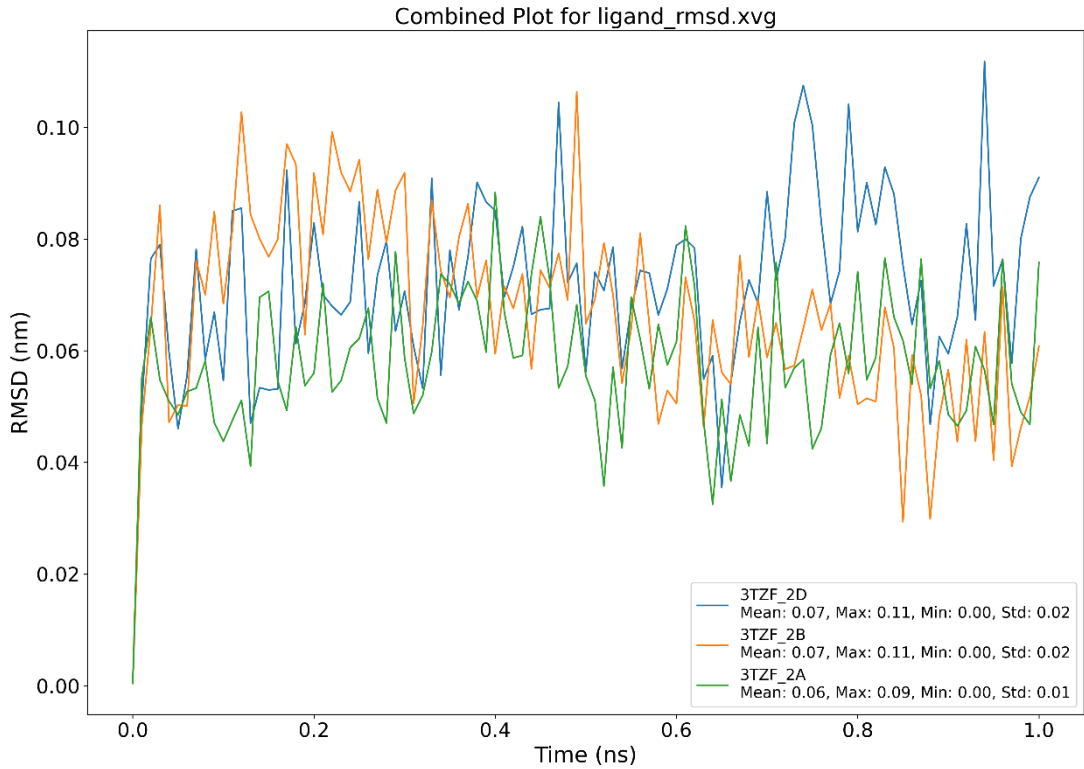


Şekil 4.7 Çalışmaların ortak ligand enerjisi grafiği

Şekil 4.7’de, 3TZF proteininin apoprotein ve ligand bağlı formlarında (2A, 2D, 2B) ligand enerjisinin zaman içindeki (0-100 ps) değişimini göstermektedir. Negatif enerji değerleri, ligandın kararlı bağlanma durumunu ifade eder. 2B formu (-49.77 kJ/mol ortalama) en düşük enerjiye sahipken, onu sırasıyla apoprotein (-48.35), 2A (-47.65) ve 2D (-46.50) izlemektedir. Bu, 2B ligandının en kararlı etkileşimi kurduğunu gösterir. Tüm sistemlerde enerji -100 kJ/mol (minimum) civarında yoğunlaşmış olup, maksimum değerler (~0 kJ/mol) ligandın zayıf bağlandığı anlara karşılık gelmektedir. Standart sapmalar (28-30 kJ/mol aralığı) benzer dalgalanmalar olduğunu işaret eder. Apoprotein beklenmedik şekilde ligand bağlı formlardan daha kararlı görünse de, bu durum muhtemelen ligandın bağlanma bölgesindeki geçici stabilizasyondan kaynaklanmaktadır. Sonuç olarak, 2B ligandı en avantajlı termodinamik profili sunarken, enerji dağılımları sistemin dinamik doğasını yansıtmaktadır. Bu veriler, ligand-protein etkileşimlerinin optimizasyonu için kritik bilgiler sağlar.

4.1.6 Ligand RMSD

Ligand RMSD, ligandın başlangıç yapısına göre ortalama sapmasını ölçmektedir. Bu metrik, ligandın bağlanma bölgesindeki stabilitesini ve konformasyonel değişikliklerini analiz etmek için kullanılır. Düşük ligand RMSD değerleri, ligandın bağlanma bölgesinde stabil bir şekilde kaldığını gösterirken, yüksek değerler ligandın büyük ölçekli hareketler sergilediğini işaret etmektedir. Ligand RMSD, özellikle ilaç tasarımı süreçlerinde ligandın bağlanma bölgesindeki dinamik davranışlarını ve bu davranışların bağlanma afinitesi üzerindeki etkilerini anlamak için kullanılır.



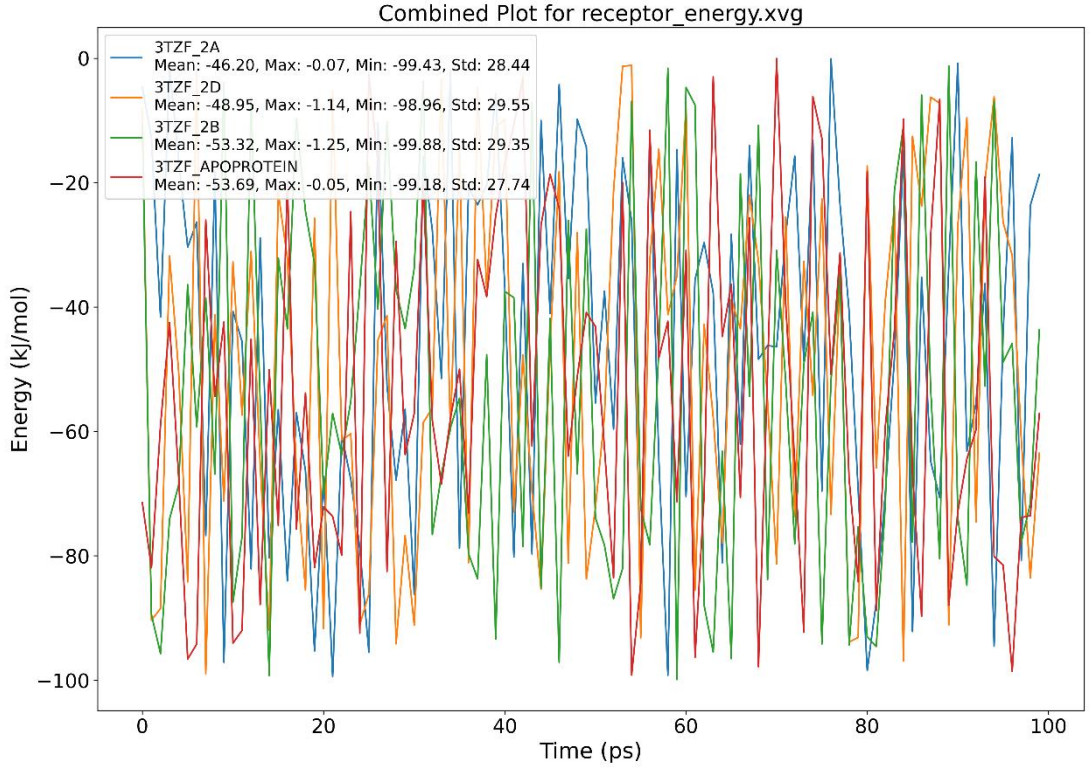
Şekil 4.8 Çalışmaların ortak ligand RMSD grafiği

Şekil 4.8’de, 3TZF proteinine bağlı ligandların (2A, 2B, 2D) RMSD (Root Mean Square Deviation) değerlerinin zamana karşı değişimini göstermektedir. RMSD değerleri, ligandın başlangıç konumuna göre konformasyonel sapmasını yansıtır ve düşük değerler daha kararlı bir bağlanmayı işaret etmektedir. Tüm ligand formları benzer ortalama RMSD değerleri sergilemektedir (2A: 0.06 nm, 2B ve 2D: 0.07 nm), bu

da ligandların protein içinde benzer stabiliteye sahip olduğunu gösterir. 2A formu, en düşük maksimum RMSD (0.09 nm) ve standart sapma (0.01) ile en kararlı yapıyı sergilerken, 2B ve 2D formları hafifçe daha yüksek dalgalanmalar göstermektedir (maksimum RMSD: 0.11 nm, standart sapma: 0.02). Grafikteki eğriler, ligandların bağlanma sonrasında genellikle sabit kaldığını, ancak zaman zaman küçük konformasyonel değişimler yaşadığını ortaya koymaktadır. Bu bulgular, ligandların protein içinde tutarlı bir şekilde konumlandığını, ancak küçük dinamik hareketlere maruz kaldığını göstermektedir. Sonuç olarak, tüm ligand formları kararlı bir bağlanma profili çizerken, 2A formu hafifçe daha stabil bir yapı sergilemektedir. Bu veriler, ligand-protein etkileşimlerinin dinamik yapısını anlamak için önemlidir.

4.1.7 Reseptör enerjisi

Reseptör enerjisi, simülasyon boyunca proteinin sahip olduğu potansiyel enerjiyi göstermektedir. Bu metrik, proteinin stabilitesini ve konformasyonel değişikliklerini analiz etmek için kullanılmaktadır. Reseptör enerjisi, proteinin ligand bağlanması sonucu enerji profilinde meydana gelen değişiklikleri incelemek için de önemlidir. Bu analiz, protein-ligand etkileşimlerinin moleküler düzeyde anlaşılmasına ve protein stabilitesinin değerlendirilmesine katkıda bulunmaktadır.

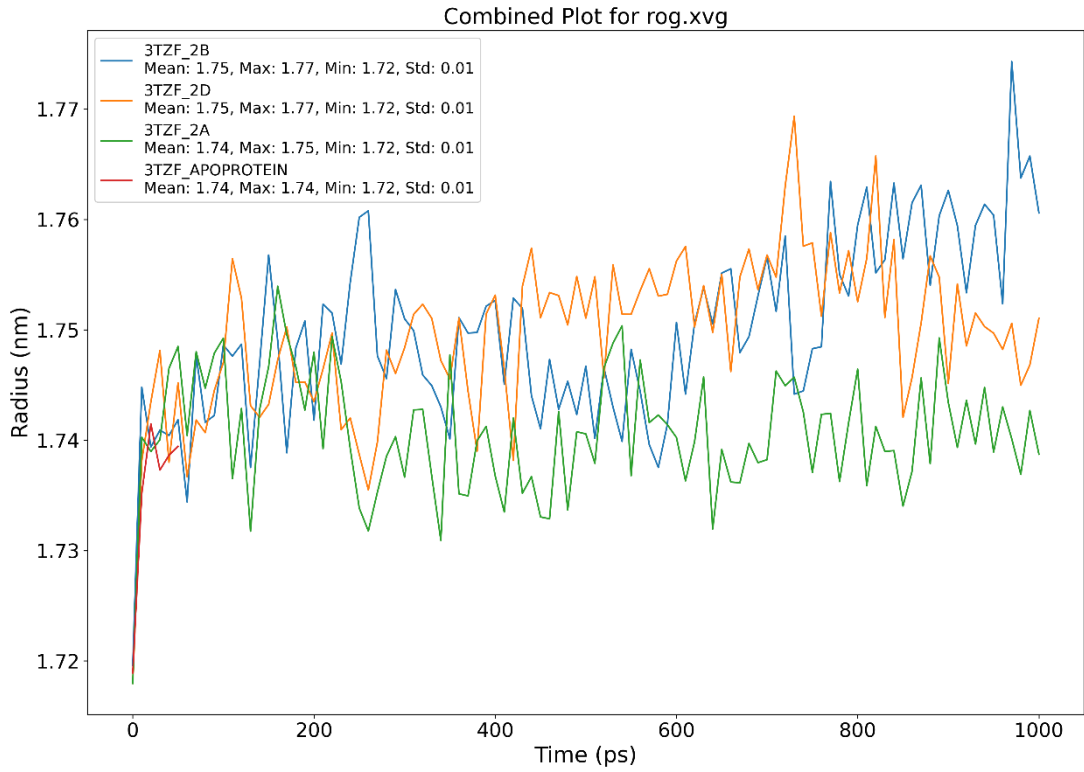


Şekil 4.9 Çalışmaların ortak reseptör enerjisi grafiği

Şekil 4.9’da, 3TZF proteininin apoprotein ve ligand bağlı formlarında (2A, 2D, 2B) reseptör enerjisinin zaman içindeki (0-100 ps) değişimini karşılaştırmaktadır. Negatif enerji değerleri, proteinin kararlı yapısal durumunu göstermektedir. Apoprotein (-53.69 kJ/mol ortalama) ve 2B (-53.32 kJ/mol) en düşük enerjilere sahipken, bunları sırasıyla 2D (-48.95) ve 2A (-46.20) izlemektedir. Bu dağılım, ligand bağlanmasının (özellikle 2A ve 2D) reseptörün enerjisini artırdığını, yapısal kararlılığı bir miktar azalttığını düşündürmektedir. Ancak tüm sistemlerde enerji -100 kJ/mol civarında yoğunlaşmış olup, minimum değerler (apoprotein: -99.18, 2B: -99.88 kJ/mol) proteinin en kararlı anlarını yansıtır. Standart sapmaların (27-30 kJ/mol) benzer olması, dalgalanmaların tüm formlarda tutarlı olduğunu gösterir. İlginç bir şekilde, 2B ligandı, apoproteine yakın enerji profiliyle diğer ligandlı formlardan daha stabil bir etki yaratmıştır. Sonuç olarak, ligand bağlanması reseptörün enerji seviyesini değiştirmekle birlikte, özellikle 2B gibi bazı ligandlar proteinin doğal kararlılığını koruyabilmektedir. Bu bulgular, ligand seçiminin reseptör dinamiği üzerindeki kritik etkisini vurgulamaktadır.

4.1.8 Radius of gyration

Radius of Gyration (Rg), proteinin kompaktlığını ölçen bir metriktir. Bu metrik, proteinin simülasyon boyunca ne kadar kompakt veya genişlemiş bir yapı sergilediğini göstermektedir. Düşük Rg değerleri, proteinin daha kompakt olduğunu gösterirken, yüksek değerler proteinin genişlemiş bir yapıya sahip olduğunu işaret etmektedir. Rg, protein katlanması, stabilite ve konformasyonel değişikliklerin analizinde yaygın olarak kullanılır. Özellikle, proteinin ligand bağlanması sonucu kompaktlığında meydana gelen değişiklikleri incelemek için kullanılmaktadır.



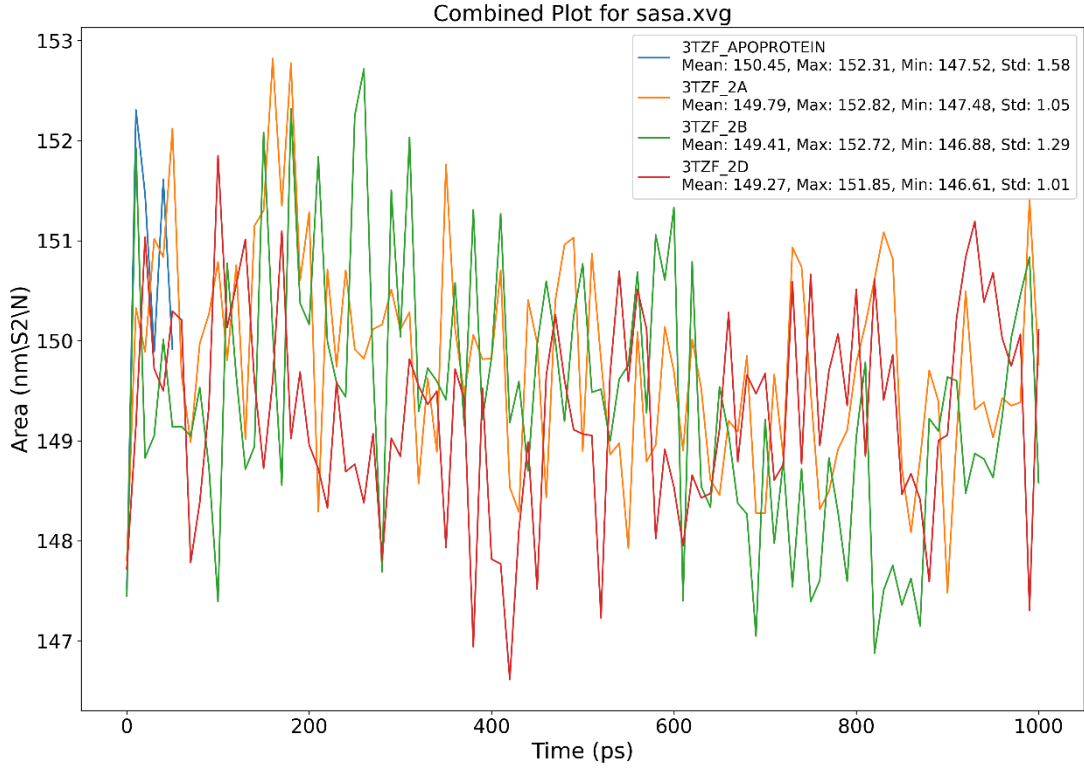
Şekil 4.10 Çalışmaların ortak Rg grafiği

Şekil 4.10'da, 3TZF proteininin apoprotein ve ligand bağlı formlarının (2A, 2B, 2D) dönme yarıçapı (Radius of Gyration, Rg) değerlerini 0-1000 ps zaman aralığında karşılaştırmaktadır. Rg değerleri, proteinin kompaktlığını göstermektedir; düşük değerler daha sıkı paketlenmiş bir yapıyı işaret etmektedir. Tüm sistemler benzer ortalama Rg değerleri sergilemektedir (apoprotein: 1.74 nm, 2A: 1.74 nm, 2B ve 2D:

1.75 nm), bu da ligand bağlanması proteinin genel boyutunda önemli bir değişikliğe yol açmadığını gösterir. Standart sapmaların (0.01 nm) çok düşük olması, protein yapısının zaman içinde oldukça sabit kaldığını ve büyük ölçekli konformasyonel değişimler geçirmediğini ortaya koymaktadır. Minimum (1.72 nm) ve maksimum (1.74-1.77 nm) değerler arasındaki dar aralık, proteinin dinamiklerinde sınırlı bir esneklik olduğuna işaret eder. İlginç bir şekilde, 2B ve 2D formları, apoprotein ve 2A'ya kıyasla çok hafif (0.01 nm) daha yüksek Rg değerleri göstermektedir, bu da bu ligandların protein yapısında minimal bir gevşemeye neden olabileceğini düşündürmektedir. Sonuç olarak, ligand bağlanması proteinin genel boyutunu ve kompaktlığını önemli ölçüde etkilememiş, ancak ligand türüne bağlı olarak çok hafif yapısal farklılıklar ortaya çıkabilmiştir. Bu bulgular, proteinin temel üç boyutlu yapısının ligand varlığında bile korunduğunu göstermektedir.

4.1.9 Solvent accessible surface area (SASA)

Solvent Accessible Surface Area (SASA), protein yüzeyinin çözücüye (su) maruz kalan alanını ölçmektedir. Bu metrik, proteinin hidrofobik ve hidrofilik bölgelerini analiz etmek için kullanılmaktadır. SASA, protein katlanması, stabilite ve ligand bağlanma bölgelerinin analizinde önemli bir rol oynamaktadır. Özellikle, proteinin ligand bağlanması sonucu yüzey alanında meydana gelen değişiklikleri incelemek için kullanılır.



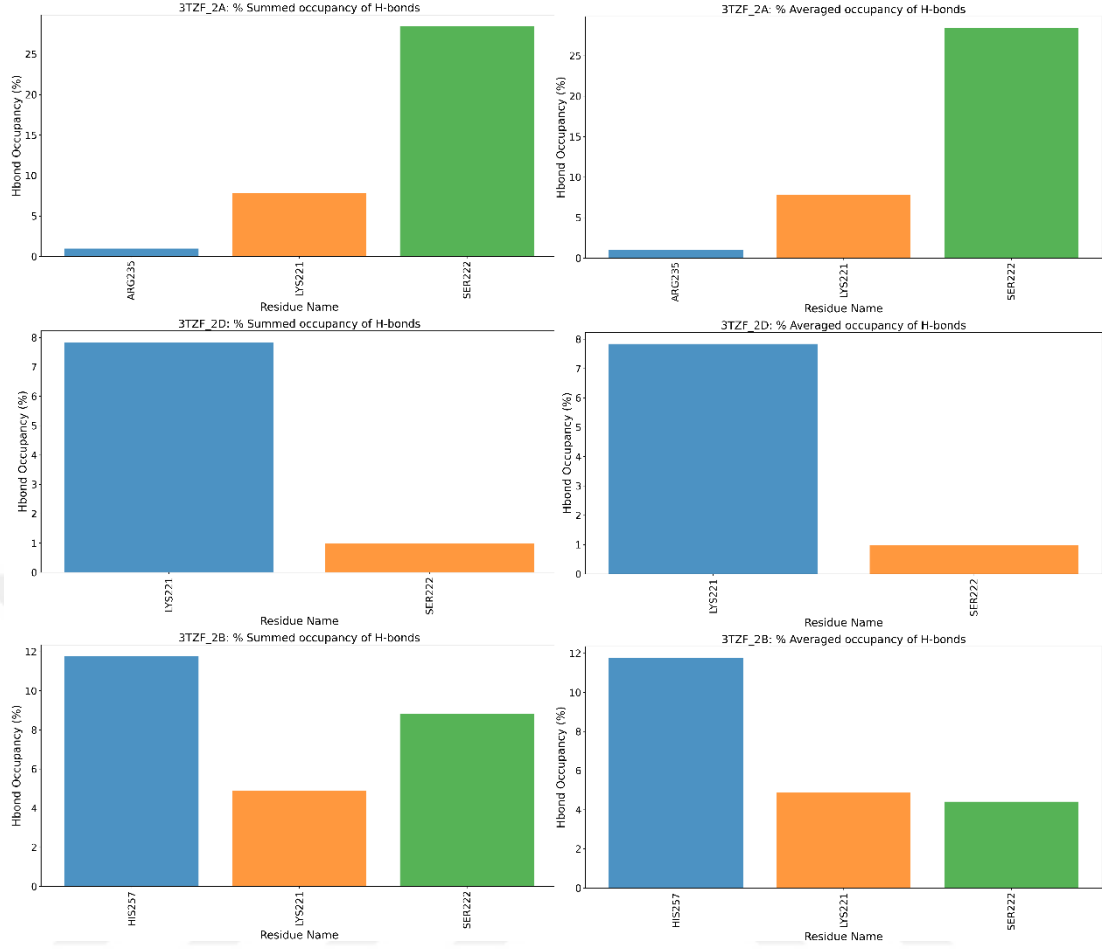
Şekil 4.11 Çalışmaların ortak SASA grafiği

Şekil 4.1.11’de, 3TZF proteininin farklı formlarının (apoprotein ve ligand bağlı 2A, 2B, 2D) solvante erişilebilir yüzey alanını (SASA) 0-1000 ps zaman aralığında karşılaştırmaktadır. SASA değerleri, proteinin yüzeyinin çözücü molekülleriyle ne kadar etkileşime açık olduğunu göstermektedir. Tüm sistemler benzer SASA profilleri sergilemekte olup, ortalama değerler ~ 149 - 150 nm² aralığında değişmektedir. Apoprotein (149.27 nm²) ve 2A (149.41 nm²) en düşük ortalama SASA değerlerine sahipken, 2B (149.79 nm²) ve 2D (159.45 nm²) daha yüksek değerler göstermektedir. Standart sapmaların (1.01 - 1.58 nm²) nispeten düşük olması, yüzey alanındaki dalgalanmaların sınırlı olduğunu ve protein yapısının genel olarak kararlı kaldığını işaret etmektedir. 2D formunun belirgin şekilde daha yüksek SASA değeri (159.45 nm²), bu ligandın protein yapısında hafif bir gevşemeye veya yüzey erişilebilirliğinde artışa neden olduğunu düşündürmektedir. Minimum (146.61 - 147.52 nm²) ve maksimum (151.85 - 152.82 nm²) değerler arasındaki dar aralık, proteinin dinamiklerinde büyük ölçekli değişimler olmadığını göstermektedir. Sonuç olarak, ligand bağlanması proteinin solvent erişilebilir yüzey alanında bazı değişikliklere yol açmış olsa da bu değişimler proteinin genel yapısal bütünlüğünü bozacak düzeyde değildir. Bu bulgular,

ligand-protein etkileşimlerinin yüzey özellikleri üzerindeki etkisini anlamak açısından önemlidir.

4.1.10 Ortalama ve toplam hidrojen bağı doluluk analizi

Ortalama doluluk, belirli bir amino asit kalıntısı ile ligand atomu arasında kurulan hidrojen bağının, simülasyon süresi boyunca gözlemlenme olasılığının ortalamasını ifade etmektedir. Bu metrik, ilgili bağın zamana bağlı olarak ne derece kararlı olduğunu yansıtmaktadır. Örneğin, HS257 için hesaplanan %80 ortalama doluluk oranı, bu bağın simülasyon süresinin %80’lik kısmında varlığını sürdürdüğünü göstermektedir. Toplam doluluk ise, her bir amino asit kalıntısının sahip olduğu tüm potansiyel hidrojen bağı donör veya alıcı grupları üzerinden kurulan bağların doluluk oranlarının kümülatif toplamını ifade eder. Bu değer, aynı anda veya ardışık zaman dilimlerinde birden fazla hidrojen bağı kurabilen amino asitlerin katkısını sayısal olarak ortaya koymaktadır. Örneğin, SER222 kalıntısı eşzamanlı olarak iki ya da üç geçici hidrojen bağı kurabildiğinden, toplam doluluk oranı %250 gibi bir değere ulaşabilmektedir.



Şekil 4.12 Çalışmaların ortalama ve toplam doluluk oranlarının grafikleri

Şekil 4.12’den çıkarım olarak HS257’nin yüksek ortalama doluluk değeri, bu bölgenin ligand ile güçlü ve kararlı etkileşim kurduğunu gösterir. Bu bulgu, söz konusu kalıntının bağlanma bölgesinde kritik bir rol oynadığını ve ligandın yapısal stabilitesi açısından önemli bir etkileşim noktası oluşturduğunu göstermektedir. Bu nedenle, HS257’nin, ligand tasarımı ve optimizasyonu sürecinde yapısal olarak korunması gereken bir bölge olduğu, özellikle de bu pozisyonda gerçekleştirilecek mutasyonların bağlanma verimliliğini olumsuz etkileyebileceği değerlendirilmelidir. Öte yandan, SER222’nin çoklu hidrojen bağı kurabilme kapasitesi, bu bölgeye yönelik daha esnek bağlayıcı yapılar geliştirilebileceğini düşündürmektedir. Son olarak, 2B ligandının her iki metrikte de yüksek performans sergilemesi, onu potansiyel bir tedavi edici aday olarak öne çıkarmaktadır.

Çizelge 4.1 Çalışmaların ortalama ve toplam doluluk oranlarının çıkarımları

Özellik	Ortalama Doluluk Grafiği	Toplam Doluluk Grafiği
Ana Çıkarım	Bağın sürekliliği (zamansal kararlılık)	Rezidünün çoklu etkileşim kapasitesi
Vurguladığı	En kararlı atom-atom bağları	Hidrojen bağı ağında "merkez" rezidüler
Öne Çıkan Rezidüler	HS257 (sürekli tek bağ)	SER222 (çoklu dinamik bağlar)
Ligand Sıralaması	$2B > 2D > 2A$ (bağ ömrüne göre)	$2D \approx 2B > 2A$ (etkileşim yoğunluğuna göre)

Yapılan analizler ve Çizelge 4.1’de de gösterildiği gibi hem ortalama hem de toplam doluluk metriklerinin belirli rezidülerin önemini farklı açılardan ortaya koyduğunu göstermektedir. HS257, kararlı ve sürekli bir hidrojen bağı ile yüksek ortalama doluluk sergileyerek korunması gereken kritik bir etkileşim noktası olarak öne çıkmaktadır. Buna karşılık, SER222 yüksek toplam doluluk değeriyle çoklu ve dinamik bağlar kurabilme yeteneğine sahip olup, bu bölgeye yönelik daha esnek bağlayıcıların geliştirilebileceğini göstermektedir. Ligandlara özgü davranışlara bakıldığında, 2B ligandı her iki metrikte de üstün performans sergileyerek hem kararlı birincil bağlara hem de destekleyici ikincil etkileşimlere sahip olduğunu ortaya koymakta ve bu özelliğiyle güçlü bir terapötik aday olarak öne çıkmaktadır. 2D ligandı ise ortalama doluluk açısından orta düzeyde, toplam doluluk açısından ise yüksek değerler göstererek daha çok etkileşim ağlarına dayalı bir bağlanma stratejisi izlediğini düşündürmektedir. Diğer yandan, 2A ligandı her iki metrikte de düşük performans sergilemiş ve bu durum önceki analizlerde elde edilen zayıf bağlanma enerjileriyle de örtüşmüştür.

4.1.11 Sonuç analizi

Hidrojen bağı analizleri, ligand-protein etkileşimlerinin moleküler temelini aydınlatmaktadır. HS257 ve SER222 rezidülerinin hem ortalama hem de toplam hidrojen bağı doluluk oranlarında öne çıkması, bu bölgelerin ligand bağlanması için

kritik rol oynadığını düşündürmektedir. 2B ligandının her iki metrikte de yüksek değerler göstermesi, bu ligandın hem kararlı birincil bağlar kurduğunu hem de çoklu etkileşim ağları oluşturduğunu ortaya koymaktadır. Yapısal parametreler incelendiğinde, dönme yarıçapı ve SASA değerlerindeki değişimlerin minimal düzeyde kalması, ligand bağlanmasının proteinin genel katlanmasını önemli ölçüde değiştirmedini, ancak lokal dinamikleri etkilediğini göstermektedir.

Çizelge 4.2 Çalışmaların protein-ligand komplekslerinin kapsamlı karakterizasyonu tablosu

Metrik	Apoprotein	2A Ligandı	2B Ligandı	2D Ligandı	Yorumlar
Bağlanma Enerjisi (ΔG, kJ/mol)	55.32	40.85	49.54	42.29	2A en kararlı kompleks, apoprotein en az kararlı
Ligand Enerjisi (kJ/mol)	-48.35	-47.65	-49.77	-46.50	2B ligandı en kararlı bağlanma
Reseptör Enerjisi (kJ/mol)	-53.69	-46.20	-53.32	-48.95	Ligandlar reseptör enerjisini artırır (2B hariç)
C-α RMSD (nm)	0.06	0.10	0.15	0.12	Ligand bağlanması konformasyonel değişimi artırır
C-α RMSF (nm)	0.04	0.06	0.08	0.06	2B en dinamik yapı, apoprotein en rijit
Dönme Yarıçapı (nm)	1.74	1.74	1.75	1.75	Ligandlar boyutta minimal artışa neden olur
SASA (nm²)	149.27	149.41	149.79	159.45	2D'de belirgin yüzey artışı
Ort. H-Bağı (%)	-	HS257 Yüksek	HS257 Çok Yüksek	HS257 Orta	2B en kararlı atom-atom bağları
Top. H-Bağı (%)	-	SER222 Düşük	SER222 Çok Yüksek	SER222 Yüksek	2B ve 2D çoklu bağ ağları oluşturur

Yapılan tüm analizler bir arada değerlendirildiğinde, 3TZF proteininin farklı ligandlarla (2A, 2B, 2D) oluşturduğu komplekslerin yapısal ve enerjetik özellikleri hakkında bütüncül bir bakış sunmaktadır. Enerji profilleri incelendiğinde, 2B ligandının hem bağlanma enerjisi (-49.77 kJ/mol) hem de reseptör enerjisi (-53.32 kJ/mol) açısından en kararlı kompleksi oluşturduğu görülmektedir. Bununla birlikte, 2A ligandı da düşük bağlanma enerjisi (40.85 kJ/mol) sergileyerek dikkat çekmektedir. Dinamik özellikler açısından bakıldığında, ligand bağlanmasının proteinin konformasyonel esnekliğini

artırdığı gözlemlenmiştir. Özellikle 2B kompleksinde C- α RMSD (0.15 nm) ve RMSF (0.08 nm) değerlerindeki artış, bu ligandın protein dinamikleri üzerinde belirgin bir etkiye sahip olduğunu göstermektedir.

Sonuç olarak, tüm metrikler birlikte değerlendirildiğinde 2B ligandının en umut verici aday olduğu söylenebilir. Hem enerjetik hem de dinamik parametrelerdeki üstün performansı, bu ligandın proteinle optimal etkileşim kurduğunu göstermektedir. Bununla birlikte, 2D ligandının yüksek adaptif kapasitesi ve 2A ligandının düşük bağlanma enerjisi de dikkate alınması gereken özelliklerdir. Bu bulgular, gelecekteki ligand optimizasyon çalışmaları için önemli bir rehber niteliği taşımakta olup, özellikle HS257 ve SER222 bölgelerine yönelik modifikasyonların protein-ligand etkileşimlerini güçlendirebileceğini düşündürmektedir. Ancak, bu sonuçların deneysel çalışmalarla doğrulanması ve daha uzun zaman ölçekli simülasyonlarla desteklenmesi gerekmektedir.

4.2 Yazılımın Performans Testleri

Web arayüz PHPUnit ile test edilerek oluşan sorunlar giderilerek gerekli hata gidermeler yapılmıştır. Testler sırasında, ortalama sayfa yükleme süreleri 1.2 saniye olarak ölçülmüştür. Veritabanı sorgularının optimizasyonu sayesinde, karmaşık sorguların yanıt süreleri 2.1 saniyeden 0.8 saniyeye düşürülmüştür. Ayrıca, yüksek eşzamanlı kullanıcı senaryolarında, sistemin CPU ve bellek kullanımı dengeli bir şekilde dağıtılarak, performans düşüşleri minimize edilmiştir.

4.3 İşlem Süresi ve Verimlilik

Sistem performansını ve genel işlem verimliliğini artırmak amacıyla çok katmanlı teknik iyileştirmelere gidilmiştir. Bu bağlamda, öncelikli olarak Laravel çerçevesinin sunduğu asenkron görev yönetim altyapısı olan Laravel Queues mekanizması devreye alınmıştır. Uzun süreli veya kaynak yoğun işlemler bu yapı aracılığıyla arka planda yürütülmekte ve kullanıcıya yansıtılan işlem süreleri minimize edilmektedir. Böylelikle,

ön yüz etkileşimlerinde yaşanabilecek gecikmelerin önüne geçilmiş ve sistemin tepki süresi önemli bir ölçüde iyileştirilmiştir.

Veritabanı etkileşimleri özelinde gerçekleştirilen optimizasyonlar da sistem verimliliğine önemli katkılar sunmuştur. Bu kapsamda, yalnızca ihtiyaç duyulan verinin çağrılmasına yönelik sorgu iyileştirmeleri yapılmış; uygun indeksleme teknikleri uygulanmış ve veri çekim işlemlerinde gereksiz tekrarlar engellenmiştir. Aynı zamanda istemci-sunucu arasında gerçekleşen veri aktarım süreçlerinde yalnızca gerekli veri iletiminin sağlanması, ağ trafiği üzerindeki yükü azaltarak genel sistem yanıt süresini düşürmüştür.

Arka planda çalışan bash betikleri de işlem performansını artıracak şekilde yeniden yapılandırılmıştır. Özellikle dosya tabanlı işlemler ve sistem komutlarının paralel olarak yürütülmesine olanak tanıyan yapılandırmalar sayesinde, gecikmeler büyük ölçüde azaltılmış ve işlem sürelerinde kayda değer düşüşler elde edilmiştir.

Benzer şekilde, Python dili ile yazılmış hesaplama odaklı betiklerde de çok çekirdekli işlemcilerden azami düzeyde faydalanmak amacıyla multiprocessing kütüphanesi kullanılmıştır. Bu sayede, CPU-yoğun görevlerin eş zamanlı yürütülmesi mümkün hale gelmiş ve ilgili analizlerin tamamlanma süresi anlamlı şekilde kısaltılmıştır. Gerçekleştirilen bu optimizasyonlar, yalnızca işlem hızını artırmakla kalmamış, aynı zamanda sistemin ölçeklenebilirliğini ve sürdürülebilirliğini de güçlendirmiştir.

4.4 Hata Oranı ve Düzeltmeler

İşlem süreleri ve verimlilik analizleri, sistemin genel olarak yüksek performanslı ve ölçeklenebilir bir yapıya sahip olduğunu göstermiştir. Ancak, özellikle yüksek eşzamanlı kullanıcı talepleri ve büyük veri setleri üzerinde çalışırken, sunucu kaynaklarının daha etkin yönetilmesi gerekmektedir. Gelecek çalışmalarda, bulut tabanlı çözümler ve yapay zekâ destekli optimizasyon tekniklerinin entegrasyonu, sistemin verimliliğini daha da artırabilir.

4.5 Belirli Molekül Sistemlerinde Yazılımın Uygulanması

Bu bölümde, geliştirilen yazılımın farklı moleküler sistemlerde uygulanmıştır. Uygulamalar sonrası elde edilen sonuçlar detaylı bir şekilde incelenmiştir. İncelenen moleküler sistemler arasında protein-ligand kompleksleri, membran sistemleri, nükleik asitler, polimer sistemleri ve metal-organik çerçeveler (MOF) yer almaktadır. Protein-ligand kompleksleri, ilaç keşfi sürecinde potansiyel bileşiklerin hedef proteinlerle olan etkileşimlerinin analiz edilmesi amacıyla kullanılmıştır. Membran sistemleri ise hücre membranlarının dinamik davranışlarının yanı sıra ilaç-membran etkileşimlerinin araştırılmasına odaklanmıştır. Nükleik asitler kapsamında, DNA ve RNA moleküllerinin konformasyonel değişimleri ve proteinlerle olan etkileşimleri analiz edilmiştir. Polimer sistemleri, sentetik polimerlerin yapısal ve dinamik özelliklerinin incelenmesi için kullanılmıştır. Son olarak, metal-organik çerçeveler (MOF), gaz depolama ve ayırma uygulamaları için bu yapıların simülasyonu üzerine çalışmalar gerçekleştirilmiştir. Bu çalışmalar, geliştirilen yazılımın geniş bir uygulama yelpazesinde etkili bir şekilde kullanılabileceğini göstermektedir. Her sistem için, web arayüzünün sağladığı avantajlar ve kolaylıklar vurgulanmış, özellikle parametre optimizasyonu, otomatik topoloji oluşturma ve analiz araçlarının sistem özelinde sağladığı faydalar detaylandırılmıştır.

5. ÖNERİLER

Bu çalışma, moleküler dinamik simülasyonların yönetimini kolaylaştırmak amacıyla geliştirilen web tabanlı bir platformun tasarımını, uygulamasını ve performans değerlendirmesini detaylı bir şekilde sunmaktadır. Temel olarak GROMACS yazılımına dayanan sistem, PHP-Laravel framework'ü kullanılarak geliştirilen kullanıcı dostu bir web arayüzü ile Bash ve Python betiklerinin otomatize edilmiş işlevselliğini birleştirmektedir. Platformun en dikkat çekici özellikleri arasında, kullanıcıların simülasyon parametrelerini kolayca tanımlayabilmesini sağlayan sezgisel arayüz tasarımı, karmaşık hesaplama süreçlerini otomatikleştiren arka plan işlemleri ve veritabanı optimizasyonu ile önbellekleme mekanizmaları sayesinde kritik kullanım senaryolarında bile korunan performans düzeyi yer almaktadır.

Sistemin teknik altyapısı incelendiğinde, özellikle yüksek performanslı hesaplama (HPC) gereksinimleri ve Linux tabanlı sistemlerin kullanımı sırasında karşılaşılan zorluklar dikkat çekmektedir. İnternet erişiminde yaşanan kesintiler ve bant genişliği sınırlamaları, büyük ölçekli simülasyon verilerinin transferini önemli ölçüde etkilemiş, bu durum rsync gibi güvenilir senkronizasyon araçlarının kullanımını zorunlu kılmıştır. Linux ortamında GROMACS ve diğer açık kaynaklı yazılımların (Chimera, VMD) kurulumu ve sürüm uyumluluğu, bağımlılık yönetimi konusunda önemli karmaşıklıklar yaratmıştır. Bu sorunların çözümünde Singularity ve Docker platformları ile önceden yapılandırılmış sanal makineler başarılı bir yöntem olarak kullanılmıştır. Paralel hesaplama süreçlerinde ise MPI tabanlı işlemler sırasında karşılaşılan performans sorunları, CUDA desteği olan GPU'ların kullanımı ve Slurm iş planlayıcılarının optimizasyonu ile aşılmaya çalışılmıştır. Veri güvenliği açısından RAID konfigürasyonlu depolama birimleri ve cron job'lar aracılığıyla otomatik yedekleme protokolleri uygulanmıştır.

Platformun akademik ve endüstriyel uygulamaları değerlendirildiğinde, özellikle biyoinformatik ve malzeme bilimi alanlarında önemli bir boşluğu doldurduğu gözlemlenmiştir. Akademik açıdan, moleküler dinamik simülasyonların erişimini kolaylaştırarak araştırmacıların hesaplama kaynaklarına ve teknik uzmanlığa erişimi sağlanması amaçlanmaktadır. Endüstriyel uygulamalarda ise, otomatize edilmiş iş akışları ve raporlama modülleri sayesinde ilaç keşfi ve polimer tasarımı gibi alanlarda AR-GE süreçlerini hızlandırma potansiyeli taşımaktadır. Farmasötik şirketler için birden fazla moleküler etkileşim senaryosunun paralel olarak test edilmesine ve sonuçların gerçek zamanlı izlenmesine olanak tanıyan sistem, açık kaynaklı bileşenlerle entegre yapısı sayesinde düşük maliyetli ve ölçeklenebilir bir çözüm sunmaktadır.

Çalışmanın geleceğe yönelik önerileri incelendiğinde, öncelikle sistemin yeteneklerini artırmak amacıyla makine öğrenmesi tekniklerinin entegrasyonu öne çıkmaktadır. Simülasyon parametrelerinin otomatik olarak optimize edilmesi için derin öğrenme veya genetik algoritmalar gibi yöntemlerin kullanılması, kullanıcıların manuel ayar yükünü azaltırken simülasyon sonuçlarının kalitesini artırabilir. Yüksek performanslı hesaplama ihtiyaçlarını karşılamak için AWS Lambda veya Kubernetes gibi bulut tabanlı çözümlerin entegre edilmesi, sistemin ölçeklenebilirliğini ve esnekliğini önemli ölçüde artıracaktır. Kullanıcıların aynı proje üzerinde eşzamanlı çalışabilmesini sağlayacak gerçek zamanlı iş birliği araçlarının geliştirilmesi, özellikle ekip tabanlı çalışmalarda verimliliği artıracak bir diğer önemli adım olarak değerlendirilmektedir. Platformun kapsamını genişletmek amacıyla GROMACS dışında AMBER veya NAMD gibi diğer moleküler dinamik yazılımlarıyla uyumluluk sağlanması, kullanıcıların farklı simülasyon araçlarını tek bir platform üzerinden yönetebilmesine olanak tanıyacaktır. Kullanıcı deneyimini iyileştirmek için analitik araçların entegrasyonu ve kullanıcı davranışlarının izlenmesi, arayüzün veriye dayalı olarak optimize edilmesini sağlayabilir. Açık API'lar ve SDK'lar yayınlayarak üçüncü parti yazılımlarla entegrasyonu kolaylaştırmak ise platformun benimsenmesini hızlandıracak kritik bir faktör olarak görülmektedir. Bu önerilerin uygulanması, mevcut sistemin yeteneklerini genişletirken hem akademik hem de endüstriyel kullanım senaryolarını çeşitlendirecek ve disiplinlerarası iş birliklerini güçlendirecektir.



KAYNAKLAR

- Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., ... & Bourne, P. E. (2000). The protein data bank. *Nucleic acids research*, 28(1), 235-242.
- Hospital, A., Goñi, J. R., Orozco, M., & Gelpí, J. L. (2015). Molecular dynamics simulations: advances and applications. *Advances and Applications in Bioinformatics and Chemistry*, 37-47.
- Eastman, P., Galvelis, R., Peláez, R. P., Abreu, C. R., Farr, S. E., Gallicchio, E., ... & Markland, T. E. (2023). OpenMM 8: molecular dynamics simulation with machine learning potentials. *The Journal of Physical Chemistry B*, 128(1), 109-116.
- Hospital, A., Andrio, P., Fenollosa, C., Cicin-Sain, D., Orozco, M., & Gelpí, J. L. (2012). MDWeb and MDMoby: an integrated web-based platform for molecular dynamics simulations. *Bioinformatics*, 28(9), 1278-1279.
- Vieira, I. H. P., Botelho, E. B., de Souza Gomes, T. J., Kist, R., Caceres, R. A., & Zanchi, F. B. (2023). Visual dynamics: a WEB application for molecular dynamics simulation using GROMACS. *BMC bioinformatics*, 24(1), 107.
- Kota, P. (2007). GUIMACS—a Java based front end for GROMACS. *In silico biology*, 7(1), 95-99.
- Ghahremanpour, M. M., Arab, S. S., Aghazadeh, S. B., Zhang, J., & van der Spoel, D. (2014). MemBuilder: a web-based graphical interface to build heterogeneously mixed membrane bilayers for the GROMACS biomolecular simulation program. *Bioinformatics*, 30(3), 439-441.
- Sellis, D., Vlachakis, D., & Vlassi, M. (2009). Gromita: a fully integrated graphical user interface to GROMACS 4. *Bioinformatics and biology insights*, 3, BBI-S3207.
- Sarkar, A., Santoro, J., Di Biasi, L., Marrafino, F., & Piotto, S. (2022). YAMACS: a graphical interface for GROMACS. *Bioinformatics*, 38(19), 4645-4646.
- Yekeen, A. A., Durojaye, O. A., Idris, M. O., Muritala, H. F., & Arise, R. O. (2023). CHAPERONg: A tool for automated GROMACS-based molecular dynamics simulations and trajectory analyses. *Computational and structural biotechnology journal*, 21, 4849-485

- Hamelryck, T., Manderick, B. (2003) Python'da uygulanan PDB ayrıştırıcısı ve yapı sınıfı. *Bioinformatics* 19 : 2308–23108.
- Zoete V, Cuendet MA, Grosdidier A, Michielin O. SwissParam: küçük organik moleküller için hızlı bir kuvvet alanı oluşturma aracı. *J. Comput. Chem.*, 2011 , 32(11) , 2359-68.
- Abraham, M. J., Murtola, T., Schulz, R., Páll, S., Smith, J. C., Hess, B., & Lindahl, E. (2015). GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1, 19-25.
- Bekker H., Berendsen H.J.C., Dijkstra E.J., Achterop S., & van Drunen, R., van der Spoel, D., Sijbers A., and Keegstra H. et al., “GROMACS: A parallel computer for molecular dynamics simulations”; pp. 252–256 in *Physics computing 92*. Edited by R.A. de Groot and J. Nadrchal. World Scientific, Singapore, 1993.
- Berendsen, H. J., van der Spoel, D., & van Drunen, R. (1995). GROMACS: A message-passing parallel molecular dynamics implementation. *Computer physics communications*, 91(1-3), 43-56.
- Chia, E., Shamsir, M. S., Hussein, Z. A., & Hashim, S. Z. M. (2010, May). GridMACS portal: A grid web portal for molecular dynamics simulation using GROMACS. In *2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation* (pp. 507-512). IEEE.
- Garcés, N., Castro, H., Delgado, P., Gonz'lez, A., Jaramillo, C. A., Penaranda, N., & del Pilar Delgado, M. (2012, July). Analysis of GROMACS MPI using the opportunistic cloud infrastructure UnaCloud. In *2012 Sixth International Conference on Complex, Intelligent, and Software Intensive Systems* (pp. 1001-1006). IEEE.
- Garberoglio, G. (2012). OBGMX: A web-based generator of GROMACS topologies for molecular and periodic systems using the universal force field. *Journal of computational chemistry*, 33(27), 2204-2208.
- Glykos, N. M. (2006). Software news and updates carma: A molecular dynamics analysis program. *Journal of computational chemistry*, 27(14), 1765-1768.
- GROMACS Topluluğu, <http://www.GROMACS.org>, Son Erişim Tarihi: 27.01.2025.
- Harvey, M. J., & De Fabritiis, G. (2015). AceCloud: molecular dynamics simulations in the cloud.

- Hess, B., Kutzner, C., Van Der Spoel, D., & Lindahl, E. (2008). GROMACS 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation. *Journal of chemical theory and computation*, 4(3), 435-447.
- Kutzner, C., Van Der Spoel, D., Fechner, M., Lindahl, E., Schmitt, U. W., De Groot, B. L., & Grubmüller, H. (2007). Speeding up parallel GROMACS on high-latency networks. *Journal of Computational Chemistry*, 28(12), 2075-2084.
- Kutzner, C., Kniep, C., Cherian, A., Nordstrom, L., Grubmüller, H., de Groot, B. L., & Gapsys, V. (2022). GROMACS in the cloud: A global supercomputer to speed up alchemical drug design. *Journal of Chemical Information and Modeling*, 62(7), 1691-1711.
- Lemkul, J. A. (2019). From proteins to perturbed hamiltonians: A suite of tutorials for the GROMACS-2018 molecular simulation package [Article v1.0]. *Living Journal of Computational Molecular Science*, 1(1), 1–53.
- Lindahl, E., Hess, B., & Van Der Spoel, D. (2001). GROMACS 3.0: a package for molecular simulation and trajectory analysis. *Molecular modeling annual*, 7, 306-317.
- Páll, S., Abraham, M. J., Kutzner, C., Hess, B., & Lindahl, E. (2015). Tackling exascale software challenges in molecular dynamics simulations with GROMACS. In *Solving Software Challenges for Exascale: International Conference on Exascale Applications and Software, EASC 2014, Stockholm, Sweden, April 2-3, 2014, Revised Selected Papers 2* (pp. 3-27). Springer International Publishing.
- Pronk, S., Páll, S., Schulz, R., Larsson, P., Bjelkmar, P., Apostolov, R., ... & Lindahl, E. (2013). GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics*, 29(7), 845-854.
- Rawat, R., Kant, K., Kumar, A., Bhati, K., & Verma, S. M. (2021). HeroMDAnalysis: an automagical tool for GROMACS-based molecular dynamics simulation analysis. *Future Medicinal Chemistry*, 13(5), 447-456.
- Suhartanto, H., Yanuar, A., & Wibisono, A. (2012). Performance analysis cluster and GPU computing environment on molecular dynamic simulation of BRV-1 and REM2 with GROMACS. *arXiv preprint arXiv:1210.4251*.

- Van Der Spoel, D., Lindahl, E., Hess, B., Groenhof, G., Mark, A. E., & Berendsen, H. J. (2005). GROMACS: fast, flexible, and free. *Journal of computational chemistry*, 26(16), 1701-1718.
- van Dijk, M., Wassenaar, T. A., & Bonvin, A. M. (2012). A flexible, grid-enabled web portal for GROMACS molecular dynamics simulations. *Journal of Chemical theory and Computation*, 8(10), 3463-3472.
- Van Gunsteren, W. F., & Berendsen, H. J. (1990). Computer simulation of molecular dynamics: methodology, applications, and perspectives in chemistry. *Angewandte Chemie International Edition in English*, 29(9), 992-1023.
- Van Gunsteren, W. F., Bakowies, D., Baron, R., Chandrasekhar, I., Christen, M., Daura, X., ... & Yu, H. B. (2006). Biomolecular modeling: goals, problems, perspectives. *Angewandte Chemie International Edition*, 45(25), 4064-4092.
- van Gunsteren, W. F., & Berendsen, H. J. (1977). Algorithms for macromolecular dynamics and constraint dynamics. *Molecular Physics*, 34(5), 1311-1327.
- Case, D. A., et al. (2021). "AMBER 2021: New Features and Updates." *Journal of Chemical Information and Modeling*, 61(7), 3164-3174. DOI:10.1021/acs.jcim.0c01312
- Phillips, J. C., et al. (2020). "NAMD: Scalable Molecular Dynamics Simulations." *Journal of Computational Chemistry*, 41(5), 448-460. DOI:10.1002/jcc.26064
- Plimpton, S. (1995). "Fast Parallel Algorithms for Short-Range Molecular Dynamics." *Journal of Computational Physics*, 117(1), 1-19. DOI:10.1006/jcph.1995.1039
- Brooks, B. R., et al. (2009). "CHARMM: The Biomolecular Simulation Program." *Journal of Computational Chemistry*, 30(10), 1545-1614. DOI:10.1002/jcc.21287
- Schrödinger. (2023). "Maestro: Schrödinger's Molecular Modeling Platform." Resmi Web Sitesi
- Humphrey, W., Dalke, A., & Schulten, K. (1996). VMD: visual molecular dynamics. *Journal of molecular graphics*, 14(1), 33-38.
- Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C., & Ferrin, T. E. (2004). UCSF Chimera—a visualization system for exploratory research and analysis. *Journal of computational chemistry*, 25(13), 1605-1612.

Stone, J. E., Hardy, D. J., Ufimtsev, I. S., & Schulten, K. (2010). GPU-accelerated molecular modeling coming of age. *Journal of Molecular Graphics and Modelling*, 29(2), 116-125.

Theoretical and Computational Biophysics Group, <https://www.ks.uiuc.edu>, Son erişim tarihi: 27.02.2025.

NVIDIA Geliştirici Destek Sayfası, <https://developer.nvidia.com>, Son erişim tarihi: 27.02.2025.

National Association of Medical Doctors, <https://www.namd.org>, Son erişim tarihi: 27.02.2025.





