

**T.C.
HACETTEPE ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ**

**prCell: SENTETİK BİYOLOJİ İÇİN SINYAL YOLAĞI
PROGRAMLAMAYA YÖNELİK BİR YAZILIM**

Mecit AKTAŞ

**Biyoinformatik Programı
YÜKSEK LİSANS TEZİ**

ANKARA

2019

**T.C.
HACETTEPE ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ**

**prCell: SENTETİK BİYOLOJİ İÇİN SINYAL YOLAĞI
PROGRAMLAMAYA YÖNELİK BİR YAZILIM**

Mecit AKTAŞ

**Biyoinformatik Programı
YÜKSEK LİSANS TEZİ**

**TEZ DANIŞMANI
Prof. Dr. Erdem KARABULUT**

ANKARA

2019

HACETTEPE ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ

Mecit AKTAŞ

prCell: SENTETİK BİYOLOJİ İÇİN SİNYAL YOLAĞI PROGRAMLAMAYA YÖNELİK BİR
YAZILIM

Danışman: Prof. Dr. Erdem KARABULUT

Bu tez çalışması 6/09/2019 tarihinde jürimiz tarafından "Biyoinformatik Programı"
nda yüksek lisans tezi olarak kabul edilmiştir.

Jüri Başkanı: **Doç. Dr. Yeşim AYDIN SON** 
(Ortadoğu Teknik Üniversitesi)

Tez Danışmanı: **Prof. Dr. Erdem KARABULUT** 
(Hacettepe Üniversitesi)

Üye: **Dr. Öğr. Üyesi Ceren SUCULARLI** 
(Hacettepe Üniversitesi)

Bu tez Hacettepe Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin
ilgili maddeleri uyarınca yukarıdaki jüri tarafından uygun bulunmuştur.

27 Eylül 2019



Prof. Dr. Diclehan ORHAN

Enstitü Müdürü

YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe Üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarında (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan "**Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge**" kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H.Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- o Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir. ⁽¹⁾
- o Enstitü / Fakülte yönetim kurulunun gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren ... ay ertelenmiştir. ⁽²⁾
- o Tezimle ilgili gizlilik kararı verilmiştir. ⁽³⁾

27/09/2019



Mecit Aktas

¹"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"

- (1) Madde 6. 1. Lisansüstü teze ilgili patent başvurusu yapılması veya patent alma sürecinin devam etmesi durumunda, tez **danışmanın** önerisi ve **enstitü anabilim dalının** uygun görüşü üzerine **enstitü** veya **fakülte yönetim kurulu** iki yıl süre ile tezin erişime açılmasının ertelenmesine karar verebilir.
- (2) Madde 6. 2. Yeni teknik, materyal ve metotların kullanıldığı, henüz makaleye dönüşmemiş veya patent gibi yöntemlerle korunmamış ve internetten paylaşılması durumunda 3. şahıslara veya kurumlara haksız kazanç imkanı oluşturabilecek bilgi ve bulgular içeren tezler hakkında tez **danışmanın** önerisi ve **enstitü anabilim dalının** uygun görüşü üzerine **enstitü** veya **fakülte yönetim kurulunun** gerekçeli kararı ile altı ayı aşmamak üzere tezin erişime açılması engellenebilir.
- (3) Madde 7. 1. Ulusal çıkarları veya güvenliği ilgilendiren, emniyet, istihbarat, savunma ve güvenlik, sağlık vb. konulara ilişkin lisansüstü tezlerle ilgili gizlilik kararı, **tezin yapıldığı kurum** tarafından verilir *. Kurum ve kuruluşlarla yapılan işbirliği protokolü çerçevesinde hazırlanan lisansüstü tezlere ilişkin gizlilik kararı ise, **ilgili kurum ve kuruluşun önerisi** ile **enstitü** veya **fakültenin** uygun görüşü üzerine **üniversite yönetim kurulu** tarafından verilir. Gizlilik kararı verilen tezler Yükseköğretim Kuruluna bildirilir.
- Madde 7.2. Gizlilik kararı verilen tezler gizlilik süresince enstitü veya fakülte tarafından gizlilik kuralları çerçevesinde muhafaza edilir, gizlilik kararının kaldırılması halinde Tez Otomasyon Sistemine yüklenir

* Tez **danışmanın** önerisi ve **enstitü anabilim dalının** uygun görüşü üzerine **enstitü** veya **fakülte yönetim kurulu** tarafından karar verilir.

ETİK BEYAN

Bu alıřmadaki bütn bilgi ve belgeleri akademik kurallar erevesinde elde ettiđimi, grsel, iřitsel ve yazılı tm bilgi ve sonuları bilimsel ahlak kurallarına uygun olarak sunduđumu, kullandıđım verilerde herhangi bir tahrifat yapmadıđımı, yararlandıđım kaynaklara bilimsel normlara uygun olarak atıfta bulunduđumu, tezimin kaynak gsterilen durumlar dıřında zgn olduđunu, Prof. Dr. Erdem KARABULUT danıřmanlıđında tarafımdan retildiđini ve Hacettepe niversitesi Sađlık Bilimleri Enstits Tez Yazım Ynergesine gre yazıldıđını beyan ederim.



Mecit Aktař

TEŐEKKÜR

Bu alıőmanın gerekleőtirilmesinde, iki yıl boyunca deęerli bilgilerini bizlerle paylaőan, gelecekteki mesleki hayatımda da bana verdięi deęerli bilgilerden faydalanacaęımı dűőündűđüm saygıdeęer danıőman hocam; Prof.Dr. Erdem KARABULUT'a, alıőmamda konu, kaynak ve yĐntem aısından bana sűrekli yardımda bulunarak yol gĐsteren Dr. Őęr. Őyesi Ceren Sucularlı'ya ve yűksek lisans eęitimi boyunca derslerime giren hocalarıma sonsuz teőekkűrlerimi sunarım.

Ankara, Eylűl, 2019

Mecit Aktaő

ÖZET

Aktaş, M., prCell: Sentetik Biyoloji İçin Sinyal Yolağı Programlamaya Yönelik Bir Yazılım, Hacettepe Üniversitesi Sağlık Bilimleri Enstitüsü Biyoinformatik Programı Yüksek Lisans Tezi, Ankara, 2019. Sentetik biyoloji, mühendisliğin temel ilkelerinin biyolojiye, özellikle de hücreye uygulanmasına yönelik yeni ve disiplinlerarası bir araştırma alanıdır. Yapay biyolojik sistemler tasarlamayı ve üretmeyi amaçlar. Sentetik biyoloji, hücresel işlevleri değiştirmek veya düzeltmek, hücrenin çevresel koşullara yanıt vermesini sağlamak veya hücresel gelişimi yönetmek için tasarlanıp uygulanabilen yeni sinyal yollarını tasarlamak veya ayarlamak için çok önemlidir. Araştırmacıları biyolojik devre tasarımının elektronikteki entegre devrelerin tasarımına benzeyeceği ve genetik mühendisliğinin yeteneklerini artıracığı bir gelecek bekliyor. Bir hücre programlama dili olarak geliştirilen uygulamaların amacı, canlı hücreler için genetik devreler tasarlamak ve hücrede tasarlanmış genetik devreyi gerçekleştirmektir. Bunlardan biri, elektronik devre tasarım programına dayanan, verilog olan ve elektronik devreleri tasarlamak için mantık kapılarını kullanan Cello'dur. Elektronik devre tasarım mantığının biyolojik devrelere uygulanmasının yeterli olamayacağını düşündüğümüzden, bu tezde prCell adını verdiğimiz biyolojik devrelerin tasarımı için yeni bir programlama dili geliştirdik. prCell'in ana özelliği, herhangi bir elektronik devre tasarımında biyolojik devreyi tasarlamak için kullanılan mantık geçitleri yerine, kullanıcının giriş-çıkış ve kontrol nesnelerini belirleyeceği dinamik yapılara dayanan bir mantığa sahip olmasıdır. Bu, çok büyük ve karmaşık bir biyolojik devre tasarlama esnekliği ve imkanı sağlar. İnternet üzerinden yayınlanacak olan prcell'de bir üyelik sistemi olacağından, kullanıcının bir veritabanı ve kullanıcının bir biyolojik devre tasarım ortamı olacaktır. En önemli hedefimiz sentetik biyoloji ile ilgilenen herkesin sinyal yolunu kolay ve etkili bir şekilde tasarlayıp canlı hücrede uygulayabilmesidir. Hücre içi olayları daha iyi modellemek için, prcell'in kullanıcı taleplerine göre geliştirileceğine ve canlıları yeniden programlamak için farklı bir bakış açısı sağlayacağına inanıyoruz.

Anahtar kelimeler: Sentetik Biyoloji, Genetik Devreler, Hücre Programlama

ABSTRACT

Aktaş, M., prCell: A Software for Pathway Programming for Synthetic Biology, Hacettepe University Graduate School of Health Sciences, Department of Bioinformatics Master's Thesis, Ankara, 2019. Synthetic biology is a new and interdisciplinary field of research aimed at applying the fundamental principles of engineering to biology, especially to the cell. It aims to design and produce artificial biological systems. Synthetic biology is very important to for design or adjust new signaling pathways that can be designed and implemented to alter or correct cellular functions, enable the cell to respond to environmental conditions, or manage cellular development. Researchers expect a future in which the design of biological circuits will resemble the design of integrated circuits in electronics and increase the capabilities of genetic engineering. The purpose of the applications developed as a cell programming language is to design genetic circuits for living cells and to realize the designed genetic circuit in the cell. One of these is Cello, which is based on the electronic circuit design program, verilog, and uses logic gates to design electronic circuits. Since we think that the application of electronic circuit design logic to biological circuits cannot be enough, in this thesis we have developed a new programming language for designing biological circuits which we call prCell. The main feature of prCell is that instead of the logic gates used for any electronic circuit design to design the biological circuit, it has a logic based on the dynamic structures in which the user will determine the input-output and control objects. This provides flexibility and possibility to design a very large and complex biological circuit. Since there will be a membership system in prcell to be published over the internet, there will be a database of the user and a biological circuit design environment of the user. Our most important goal is that anyone interested in synthetic biology can easily and effectively design the signal pathway and apply it in the living cell. We believe that in order to better model intracellular events, prcell will be developed according to user demands and will provide a different perspective for reprogramming living things.

Key Words: Synthetic biology, Genetic circuits, Cell programming

İÇİNDEKİLER

ONAY SAYFASI	iii
YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI	iv
ETİK BEYAN	v
TEŞEKKÜR	vi
ÖZET	vii
ABSTRACT	viii
İÇİNDEKİLER	ix
SİMGELER VE KISALTMALAR	xii
ŞEKİLLER	xiii
1.GİRİŞ	1
2.GENEL BİLGİLER	3
2.1. Bileşik	3
2.2. Protein	3
2.3. Modifiye Protein	3
2.4. Protein Kompleksi	3
2.5. Biyokimyasal Reaksiyonlar	4
2.6 .Gen İfadesi	4
2.7. Sinyal yolağı	5
2.8. Sinyal yolağı Çizim Programları	5
2.9. Sentetik Biyoloji ve Genetik Devreler	6
2.10. “prCell” Sinyal Yolağı Programlama Dili	8
3. GEREÇ VE YÖNTEM	9
3.1. Nesnelere ve Etkileşimler	9
3.1.1. Sinyal Yolağında Kullanılan Temel Nesnelere	9
3.1.2. Etkileşimler	11
3.1.3. Gen İfadesi ve Biyokimyasal Reaksiyonların Modellenmesi	14
3.1.4. Bağlanma	15
3.2. İş Akışı	17
3.2.1. Üyelik	17
3.2.2. Temel Nesnelere Kaydedilmesi	17

3.2.3. Nesne ve Entity Yapısına Port Eklenmesi	17
3.2.4. Proje Oluřturma ve Diagram Çizme	19
3.3. Yazılımın Teknik Özellikleri	20
3.3.1. Veri tabanı dili	20
3.3.2. Programlama dili	21
3.3.3. Yazılım Mimarisi	21
3.3.4. Javascript Kütüphanesi	22
3.4. Veri tabanı Tasarlama	23
3.4.1. Temel Nesnelere İlgili Tablolar	23
3.4.2. Nesnelerin Etkileşim Tabloları	26
3.4.3. Entity Tabloları	28
3.4.4. Geçici ve Gerçek tablolar	31
3.4.5. Diagram Tablosu	31
3.5. Nesnelerin Modellenmesi	32
3.5.1. Temel Nesnelerin Diagramda Gösterim Şekilleri	32
3.6. Gen İfadesi ve Biyokimyasal Reaksiyonlar	33
3.6.1. Entity Modellemesi	33
3.7. Üyelik ve Temel Veri Kaydı	34
3.7.1. Üyelik	34
3.7.2. Kütüphane Oluřturma	34
3.7.3. Kompleks Yapı Oluřturma	39
3.8. Port Ekleme	41
3.8.1. Nesnelere Giriş ve Kontrol Portlarının Eklenmesi	41
4.BULGULAR	48
4.1. Tasarım	48
4.1.1. Proje Oluřturma	48
4.1.2. Diagram Çizme	49
4.2. Örnek Bir Uygulama	53
4.2.1. Kütüphaneye Veri Kaydı	53
4.2.2. Nesnelere Giriş Ve Kontrol Portlarının Eklenmesi	58
4.2.3. Proje Oluřturma ve Diagram Çizme	62
4.2.4. Proje İmport ve Export Etmek	63

4.3. Ecoli Bakterisine Ait Sinyal Yolađı Üzerine Bir Uygulama

64

5.TARTIŞMA

68

6.SONUÇ VE ÖNERİLER

70

7.KAYNAKLAR

71

8.EKLER

EK-1: Etkileşim Kuralları

EK-2: KEGG Pathway Map

EK-3: Pathvisio Sinyal Yolađı Çizim Ekranı

EK-4: Dijital Makbuz

9.ÖZGEÇMİŞ

SİMGELER ve KISALTMALAR

act	Activation
Asp	Active Server Page
Bknz	Bakınız
DNA	DeoksiriboNükleik Asit
Exp	Example
exp	Expression
HTML	Hiper Metin İşaretleme Dili (Hypertext Markup Language)
http	Hiper Metin Aktarım Protokolü (HyperText Transfer Protocol)
ID	Tanımlayıcı (IDentifier)
ind.eff	Indirect Effect
inh	İnhibition
JSON	JavaScript Object Notation
MIT	Massachusetts Institute of Technology
mRNA	Mesajcı RNA (Messenger RNA)
MSSQL	Microsoft Yapılandırılmış Sorgu Dili Sunucusu (Microsoft Structured Query Language Server)
MVC	Model, Görüntü, Kontrolcü (Model View Controller)
prCell	Programming Cell (Hücre Programlama)
RNA	RiboNükleik Asit
repres	Repression
snRNA	Küçük nükleer RNA (Small nuclear RNA)
SQL	Yapılandırılmış Sorgu Dili (Structured Query Language)
st.cha	State chance
SVG	Scalable Vector Graphics
tRNA	Taşıyıcı RNA (Transfer Ribonucleic Acid)

ŞEKİLLER

Şekil	Sayfa
2.1. Örnek devre tasarımı.	6
2.2. AND gate gösterimi.	7
2.3. OR gate gösterimi.	7
2.4. NOT gate gösterimi.	7
3.1. Bileşiklerin model gösterimi.	9
3.2. Proteinlerin model gösterimi.	10
3.3. Protein kompleks yapılarının modellenmesi.	10
3.4 Entity yapısının model gösterimi	11
3.5. Protein-protein etkileşiminin model gösterimi.	12
3.6. Protein-kompleks etkileşiminin model gösterimi .	13
3.7. Bileşik-protein etkileşiminin model gösterimi.	13
3.8. Bileşik-protein kompleksi etkileşiminin model gösterimi.	14
3.9. Bileşik-bileşik etkileşiminin model gösterimi.	14
3.10. Biyokimyasal reaksiyonların model gösterimi.	15
3.11. Gen ifadesinin model gösterimi.	15
3.12. Eşleşme kuralının gösterimi.	16
3.13. Giriş-çıkış kuralının gösterimi.	16
3.14. Bileşiklere port eklenmesinin model gösterimi.	18
3.15. Proteinlere port eklenmesinin model gösterimi.	18
3.16. Protein komplekslerine port eklenmesinin model gösterimi.	19
3.17. Entity yapılarına port eklenmesinin model gösterimi.	19
3.18. Diagram çizim ekranının model gösterimi.	20
3.19. Bileşik/molekül tablosu.	23
3.20. Protein tablosu.	24
3.21. Protein modifikasyon tür tablosu.	25
3.22. Protein kompleks tablosu.	25
3.23. Protein kompleks yapılarının bileşen tablosu.	26
3.24. Etkileşim türleri tablosu.	26
3.25. Protein-protein etkileşim tablosu.	27

3.26. Bileşik-protein kompleks etkileşim tablosu.	27
3.27. Entity tablosu.	29
3.28. Entity-bileşik input tablosu.	29
3.29. Entity-protein input tablosu.	30
3.30. Entity-protein output tablosu.	30
3.31. Entity-protein kontrol tablosu.	30
3.32. Diagram tablosu.	31
3.33. prCell yazılımında bileşik/molekül gösterimi.	32
3.34. prCell yazılımında protein gösterimi.	32
3.35. prCell yazılımında protein kompleksi gösterimi.	33
3.36. prCell yazılımında entity gösterimi.	33
3.37. prCell yazılımında giriş ekranı.	34
3.38. prCell yazılımında temel entity kaydı.	35
3.39. prCell yazılımında bileşik/molekül temel bilgilerinin kaydı.	36
3.40. prCell yazılımında proteinlerin temel bilgi kaydı.	37
3.41. prCell yazılımında modifiye olmuş proteinlerin kaydı.	38
3.42. prCell yazılımında protein komplekslerinin temel bilgilerinin kaydedilmesi.	39
3.43. prCell yazılımında protein komplekslerinin bileşenlerinin oluşturulması.	40
3.44. prCell yazılımında bileşik/molekül port ekleme.	42
3.45. prCell yazılımında bileşik/molekül port eklenmemiş hali.	42
3.46. prCell yazılımında bileşik/molekül.	43
3.47. prCell yazılımında proteinlere port ekleme.	43
3.48. prCell yazılımında port eklenmemiş protein.	43
3.49. prCell yazılımında giriş portu olarak M2 molekülü eklenmiş P2 proteini.	44
3.50. prCell yazılımında protein kompleklerine port eklenmesi.	44
3.51. prCell yazılımında port eklenmemiş protein kompleksi.	45
3.52. prCell yazılımında port eklenmiş C1 kompleksi.	45
3.53. prCell yazılımında entity yapısına port eklemek.	46
3.54. prCell yazılımında entity'ye port eklenmesi.	47
3.55. Kontrol portları olan bir entity yapısı.	47
4.1. prCell yazılımında temel bilgilerle proje kaydı.	48
4.2. prCell yazılımında tasarım ekranı.	49

4.3.	prCell yazılımında tasarım ortamına entity atmak.	50
4.4.	prCell yazılımında tasarım alanına eklenmiş entity.	50
4.5.	prCell yazılımında tasarım ortamına protein atmak.	51
4.6.	prCell yazılımında giriş portu olmayan P1proteini eklenmiş.	51
4.7.	prCell yazılımında tasarım ortamına protein kompleksi atmak.	52
4.8.	prCell yazılımında tasarım ortamında oluşturulmuş C2 kompleks yapısı.	52
4.9.	prCell yazılımında tasarım ortamına bağlanma kuralının uygulanması.	53
4.10.	prCell yazılımında E1 entity'sinin kaydedilmesi.	54
4.11.	prCell yazılımında M1 molekülünün/bileşiğinin kaydedilmesi.	55
4.12.	prCell yazılımında P1 proteininin kaydedilmesi.	55
4.13.	prCell yazılımında P2 proteininin P3 tarafından modifiye edilmesinin kaydedilmesi.	56
4.14.	prCell yazılımında C1 kompleks yapısının kaydedilmesi.	56
4.15.	prCell yazılımında C3 protein kompleksinin bileşenlerinin oluşturulması.	57
4.16.	prCell yazılımında oluşturulan C3 kompleksinin gösterilmesi.	57
4.17.	prCell yazılımında M1 molekülüne/bileşiğine P1,M2 ve C1 portlarının eklenmesi.	58
4.18.	prCell yazılımında oluşturulan M1 molekülünün/bileşiğinin gösterilmesi.	58
4.19.	prCell yazılımında P2 proteinine M3,M4 ve P1 portlarının eklenmesi.	59
4.20.	prCell yazılımında eklenen P2 proteininin gösterilmesi.	59
4.21.	prCell yazılımında C1 kompleksine P4,M2 ve C2 portlarının eklenmesi.	60
4.22.	prCell yazılımında C1 kompleksinin gösterimi.	60
4.23.	prCell yazılımında E1 entity'si.	61
4.24.	prCell yazılımında E1 entity'sinin gösterimi.	62
4.25.	prCell yazılımında proje kaydı girme.	62
4.26.	örnek bir yapay sinyal yolağı uygulaması.	63
4.27.	prCell yazılımında proje import etmek.	63
4.28.	prCell yazılımında projeyi export etmek.	64
4.29.	Keggpathway'den alınmış ecoli bakterisine ait bir sinyal yolağı.	65
4.30.	prCell de şekil 4.29 de gösterilen ecoli bakterisine ait sinyal yolağının gösterimi	66

1.GİRİŞ

Sinyal yolları, bir hücredeki moleküller arasında belirli bir ürüne veya hücrede bir değişikliğe yol açan ardışık olayların ve işlemlerin tamamını kapsayan sistemin bir modellenmesidir.[1] Bu işlem yolları, protein gibi yeni moleküllerin başka yapılarla birleşerek işlevsel olarak çeşitlenmesine sebep olur. Yollar ayrıca genleri etkin ya da pasif hale getirebilir. Hücre içi yaygın olaylardan bazıları metabolizma, gen ifadesinin düzenlenmesi ve sinyallerin iletimi ile ilgilidir. Geçiş yolları, genomik çalışmalarında kritik rol oynar.[2]

Sinyal yolları hücrede hücresel aktiviteleri modelleyen, hücrenin tepkisini düzenleyen, hücre iletişimi içindeki kompleks sisteminin bir ifade şeklidir. Hücrenin bilgi iletimindeki eksikler ve yanlışlar, kanser ve diyabet gibi hastalıkların önemli nedenidir. Sinyal yolları, metabolizmadaki işlevler, genetik mutasyonlar ve kanser hücrelerinin çevresindeki olaylardan etkilenmesinden dolayı kanser hücre biyolojisinin önemli bir araştırma alanı olmuştur.[3] Hücre sinyal yolları detayları ile öğrenildiğinde, henüz tedavisi olmayan hastalıkların iyileştirilmesi mümkün olacaktır

Başta kanser olmak üzere önemli hastalıkların sinyal yollarındaki hataların tespit edilmesi ve sinyal yolağı modeli üzerinde çözüm üretilebilmesi tedavi sürecini etkin ve zaman açısından da verimli kılacaktır.[4] Şu ana kadar biyolojik veritabanlarında varolan canlıların varolan sinyal yolları saklanmaktadır. Ancak bir sinyal yolağının yapay sinyal yolağı tasarlanabilen bir yazılımla yeniden tasarlanabilmesi, bu yolağın üzerinde sanal ortamda çözüm üretilebilmesini imkanı hale getirecektir.

prCell adını verdiğimiz bu tez çalışmasında yapay olarak sinyal yolağı tasarlamayı amaç edindik. Bunun en önemli iki sebebinden ilki mevcut sinyal yollarını tekrar çizerek üzerinde değişiklikler yapmak ve hastalık oluşturan kısımlara yeni yollar bulmak. Diğer amacı ise bir canlıdaki bir özelliğin başka canlılara entegre edebilmek. Her iki canlının da ilgili sinyal yolağını prCell üzerinde tekrar oluşturarak aktarmak istediğimiz özelliklerle ilgili yazılım üzerinde çalışmalar ve denemeler yapmak. Teorik olarak bulunabilecek bir yöntem pratikte uygulanabilir mi bilemeyiz ancak yazılım üzerinde dahi olsa bir çözüm bulabilmek çok değerli olacaktır.

Son amacı ise tamamıyla hiç olmayan yapay sinyal yolları oluşturmak. Teorik olarak doğru olacak olan bu sinyal yolları belki de ileride metabolizmayı tamamıyla dışarıdan tasarlanabilir ve müdahale edilebilir hale getirecektir. Bu iddali amaçları gerçekleştirebilmek

için yapılmış olan bu yazılımda, řu aşamada temel kurallara göre yapay yolaklar tasarlanabiliyor olması bizim için önemli bir eřiđin atlanması anlamına gelmektedir. Daha karmařık ve fonksiyonel biyolojik devrelerin tasarlanması için her yazılım gibi prCell'in de geliřtirilmesi ve büyütülmesi gerecektir.

2.GENEL BİLGİLER

2.1.Bileşikler

Kimya, biyokimya ve biyolojide karşımıza çıkan yapılardan biri olan bileşik; küçük moleküler yapıları, biyopolimerler ve biyolojik sistemlerle ilgili diğer kimyasal maddeleri içeren yapılardır[5].Biyolojik açıdan bileşik olarak biyomolekülleri de kabul edebiliriz. Canlı hücrelerde kendiliğinden oluşan bu moleküller birincil ve ikincil metabolitler ve doğal ürünler gibi küçük yapıları içerir. Azot, oksijen, kükürt, fosforlu karbon ve hidrojen içeren bu moleküller arasında nükleik asit, yağ ve karbonhidrat gibi makromoleküller de bulunur.[6]

2.2. Proteinler

Genlerin bir ifadesi olarak karşımıza çıkan ve amino asitlerden oluşan proteinler, canlılardaki en etkili makromoleküllerdir ve biyolojik işlemlerde çok önemli fonksiyonlara sahiptir[7]. Enzim olarak katalizör işlev görürken aynı zamanda oksijen gibi diğer molekülleri taşır ve depolar. Büyümeyi ve farklılaşmayı kontrol ederler. [8]

Bir proteinin işlevi, proteinin üç boyutlu yapısına bağlıdır. Proteinler kendiliğinden, protein polimerindeki aminoasit dizisi tarafından belirlenen üç boyutlu yapıya katlanmaktadır. Bu açıdan, proteinler, tek boyutlu kod diziliminden, çeşitli hücrenel aktiviteleri yapabilen üç boyutlu moleküller haline evrilmenin bir düzenlemesidir.[9]

2.3. Modifiye Proteinler

Proteinler sentezlikten sonra çoğu zaman bazı değişimlere uğrayabilir. Bu değişiklikler protein yapısını, substrat alanını, aktivesini ve hücre içindeki yerini etkiler. Genetik ve kimyasal yöntemlerle,yapay veya doğal yapıların kullanılmasıyla proteinler işlevsellik açısından çok çeşitlilik kazanmıştır.Bu çeşitlilik doğadaki biyoçeşitliliği doğurmuştur.Buna sebebiyet veren modifikasyonlardan bazıları arasında asilasyon, metilasyon, fosforilasyon, sülfatlama, farnesilasyon, ubikatinasyon ve glikosilasyon yer alır.[10]

2.4. Protein Kompleksleri

Proteinler, kompleks yapılar oluşturmak için birbirleriyle ve diğer biyolojik makromoleküllerle etkileşime girebilirler. Ayrı ayrı proteinler tarafından oluşmayan

yetenekler, bu proteinler çoğaltması, hücre içi sinyallerin iletimi gibi birçok önemli işlemi gerçekleştirirler.[11]

Bazı proteinler oldukça katı-sert, bazıları ise sınırlı bir esnekliğe sahiptir. Sınırlı esnekliğe sahip protein yapıları, protein işlevi için kritik önem taşıyan proteinlerin birbirleriyle ve diğer moleküller ile kompleks birimler halinde birleştirilmesi ve hücrelerin içinde ve arasında bilgi iletimi gibi davranabilirler.[12] Bazı protein kompleksleri farklı genler tarafından kodlanan birçok farklı protein alt biriminden oluşur.[13]

2.5. Biyokimyasal Reaksiyonlar

Biyokimya, canlı organizmalarda oluşan kimyasalların ve canlıda yaşam süreçlerinin araştırılmasıdır. Biyolojik olarak aktif moleküllerin sentezi gibi biyomoleküllerin işlev ve yapısının arkasındaki kimyaya incelemektedir. [14] Bir yönüyle belli inputların enzimler vasıtasıyla belli outputlara dönüşümü olarak da ifade edilebilir.

Hedeflenen bir outputu üretebilmek için sentez sırasında farklı input kombinasyonları kullanılır. Arka arkaya devam eden kimyasal reaksiyonlarda bir reaksiyonun outputu bir sonrakinin inputu olacağından metabolik yollar oluşur. Bu reaksiyonlar çoğu zaman protein enzimleriyle katalize edilir. Enzimlerin en büyük özelliği normal koşullarda imkansız veya çok zor olan metabolik sentez ve ayrışmaları hücre içinde mevcut sıcaklık ve konsantrasyonlarda, çok fazla enerji tüketmeden biyokimyasal reaksiyonların gerçekleşmesini sağlamasıdır.[15]

2.6. Gen İfadesi

Genetikte, gen ifadesi, genotipin fenotipe neden olduğu en temel seviyedir. DNA'da varolan genetik bilgi, gen ifadesiyle "yorumlanabilir" ve ifadenin özellikleri organizmanın fenotipine yol açar. Gen ifadesi, bir genden gelen bilgilerin mRNA yoluyla bir proteinin sentezine dönüşmesi sürecidir.[16] Bunun en büyük aşaması, fonksiyonel bir gen molekülü olan protein sentezine yol açan mRNA'nın sentezi ve transkripsiyonudur. Ancak transfer RNA (tRNA) veya küçük nükleer RNA (snRNA) genleri gibi protein kodlamayan genlerde, gen ifadesi işlevsel bir RNA'dır.[17] Gen ekspresyon işlemindeki süreçler: bir proteinin transkripsiyonu, RNA eklenmesi, translasyonu ve translasyon sonrası modifikasyonu olarak sıralanabilir. Gen düzenlemesi yapı ve fonksiyon üzerinde hücre kontrolünü sağlar ve hücrel değişime ve herhangi bir canlının çok yönlülüğü için temel oluşturur. Gen ifadesinin düzenlenmesi bu nedenle bir canlının gelişimi ve yaşam döngüsü için kritik bir öneme sahiptir.[18],[19]

2.7. Sinyal Yolađı

Sinyal yolađı, bir hücredeki moleküler yapılar arasındaki belirli bir çıktıya veya hücrede bir deđişikliğe yol açan bir dizi etkileşimdir. Bu durum, bir yağ veya protein gibi yeni moleküllerin oluşumunu başlatabilir. En çok karşımıza çıkan biyolojik yollardan bazıları metabolizma, gen ifadesinin düzenlenmesi ve sinyallerin iletimi ile ilgilidir.[20]

Canlılarda hücre içi düzenin korunması için bazı biyolojik yollar gereklidir. Bu sinyal yolları boyunca biyomoleküllerin akışı, hücrenin ihtiyaçlarına ve substratın durumuna göre düzenlenir. Bir biyolojik çıktı işe yarar, metabolizmaya ilişkin başka bir süreç başlatır, ya da daha sonra kullanılmak üzere hücre içinde bekletilir. Bir hücrenin metabolizması, biyokimyasal yapıların oluşumunu ve parçalanmasını sağlayan birbiriyle etkileşim halinde olan yollar ağından oluşur.[21]

Bazı proteinler sadece diđer genleri aktive etmek için vardır ve bunlar düzenleyici ağlarda veya etkileşim zincirinin ana aktörü olan transkripsiyon faktörleridir. Diđer genlerin başlangıcında promotör bölgeye bağlanarak onları açarlar, başka bir proteinin üretimini başlatırlar. Bazı transkripsiyon faktörleri ise inhibe edicidir.[22]

Sinyal yolları birbirleriyle etkileşime girdiğinde, genellikle bütünleştirici sinyal olayları ile hücre sel yanıtın koordine edilmesini sağlayan ağlar oluştururlar. Moleküler olaylar, hücre büyümesini, çođalmasını, metabolizmasını ve diđer birçok işlemi kontrol eden temel mekanizmalardır. Çok hücreli canlılarda, hücre iletimini çok çeşitli şekillerde düzenlemek için sinyal iletim yolları oluşmuştur. [23]

2.8. Sinyal Yolađı Çizim Programları

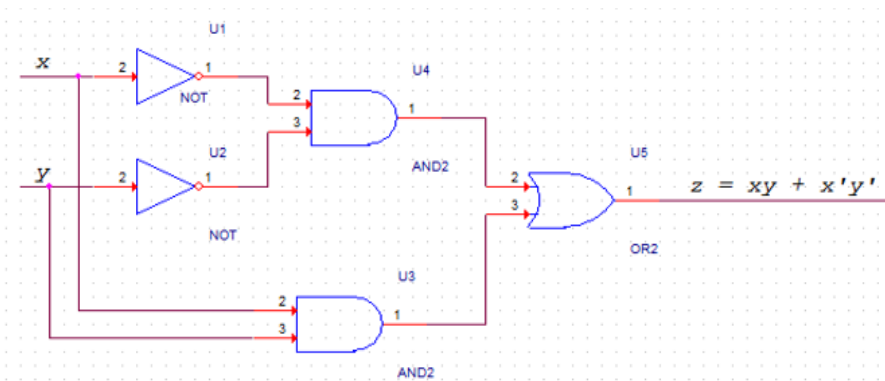
Sinyal yolađı çizmek, analiz edebilmek ve görselleştirebilmek için geliştirilmiş bazı web uygulamaları mevcuttur. Bu uygulamalar üzerinde sinyal yolađı tasarlamak mümkündür. Bir açıdan PowerPoint veya Photoshop gibi herhangi bir çizim uygulamasında olduđu gibi yolları çizmemize olanak sağlar. Çizilen yollardaki genler veya proteinlere veri tabanı bağlantısı kullanarak biyolojik bilgilerine erişebilirsiniz. En çok kullanılan uygulamalar cellillustrator[24], celldesigner[25], pathvisio[26], reactome[27] ve wikipathways[28] uygulamalarıdır. Bir diđer uygulama ise hücre programla dili olan Cello uygulamasıdır. Bu programlama dili, bilgisayar çiplerini programlamak için yaygın olarak kullanılan Verilog'u temel almaktadır.[29]

2.9. Sentetik Biyoloji ve Genetik Devreler

Sentetik biyoloji, mühendislik prensiplerinin biyoloji uygulandığı disiplinlerarası bir alandır. Bir başka ifadeyle biyolojik modüller tasarlamak, [30] biyolojik sistemler oluşturmak ve biyolojik makineler tasarlamak ve inşa etmek veya mevcut biyolojik sistemleri farklı amaçlar için yeniden dizayn etmektir. Sentetik biyoloji günümüzde sanayi veya biyolojik araştırma için uygulamaları geliştirmek amacıyla biyolojik sistemlerin ve canlı organizmaların yapay tasarımı ve mühendisliği olarak tanımlanır.[31]

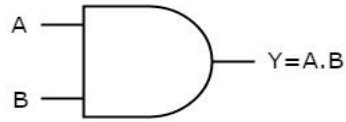
Sentetik biyoloji genel olarak iki farklı dala ayrılmıştır. Yukarıdan aşağıya sentetik biyoloji, canlı hücrelere yeni fonksiyonlar kazandırmak için metabolik ve genetik mühendislik tekniklerinin kullanılmasını içerir. Aşağıdan yukarı doğru sentetik biyoloji, 'canlı olmayan' biyomoleküler bileşikler bir araya getirilerek in vitro olarak yeni biyolojik yapılar oluşturmayı, yapay bir hücre oluşturmayı içerir. [32].

Genetik devreler, bir hücrenin içindeki biyolojik parçaların, elektronik devrelerde (**Şekil 2.1**'de gösterildiği gibi) karşılaştığımız mantıksal işlevleri yerine getirmek üzere tasarlandığı bir sentetik biyoloji uygulamasıdır. Sentetik biyolojinin amacı, istenen herhangi bir sentetik biyolojik devrenin kolayca tasarlanıp uygulanabileceği ayarlanabilir ve özellikli modül üretmektir. [33] Bu devreler hücrel fonksiyonları değiştirmek, çevresel koşullara hücrel tepkiler oluşturmak veya hücrel gelişimi etkilemek için bir yöntem olarak düşünülebilir.



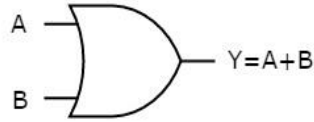
Şekil 2.1 Örnek devre tasarımı

“AND” gate kuralı girdilerin hepsinden elektrik gelirse çıktılarda da elektrik vardır (**Şekil 2.2**). Yani girdilerin hepsi “1” ise çıktı da “1”dir. Aksi durumda çıktı “0” olur. Bu durumun hücredeki karşılığı, bir proteinin üretilmesi için en az iki proteinin aktif olması gerekir. İlgili proteinlerin bir tanesi bile aktif değilse hedeflenen protein üretilemez. Yani girdide gerekli proteinlerin hepsinin de aktif olması şarttır. Bir tanesi bile olmazsa istenilen protein üretilemez demektir.



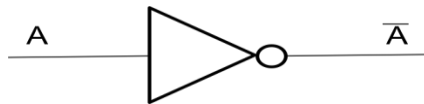
Şekil 2.2 AND gate gösterimi

Şekil 2.3’de gösterilen “OR” gate kuralında girdilerin sadece birinden elektrik gelmesi çıktıdan elektrik çıkması için yeterlidir. Yani girdilerin sadece biri “1” ise çıktı da “1” olur. Bu durumun hücredeki karşılığı ise, bir proteinin üretilmesi için en az iki proteinden sadece biri yeterli oluyorsa istenilen protein elde edilebiliyor demektir.



Şekil 2.3 OR gate gösterimi

“INVERSE” gate ise girdiden elektrik gelirse çıktıdan elektrik çıkmaz, girdiden elektrik gelmezse de çıktıdan elektrik çıkar. Yani girdi “1” ise çıktı “0”, girdi “0” ise çıktı “1” olur. Hücrede bu durum, bir proteinin üretilmesini baskılayan protein varsa istenilen protein üretilmez, baskılayıcı protein yok ise de üretilir şeklinde modellenmiştir (**Şekil 2.4**).



Şekil 2.4 NOT gate gösterimi

Ancak devrelerde kullanılan 14 mantık kapısının tamamını hücre içindeki olaylarla ilişkilendirip modellemek aşılması gereken büyük bir problem olarak karşımıza çıkmaktadır. Sınırlı sayıda mantık kapısı hücre içi durumlarla ilişkilendirilebilmiştir.

Elektronik devrelerinde kullanılan 14 mantık kapısının tamamını hücreye uyarlansa dahi hücre içindeki olayların karmaşık ve çokluğundan dolayı yeterli olmayacaktır. Bu açıdan hücre programlama dili için farklı bir mantık geliştirmek gerekmektedir.

2.10. “prCell” Sinyal Yolağı Programlama Dili

Sinyal yolağı çizim uygulamalarının programlama mantığı taşımamasından ve “Cello” programlama dilinin de hücredeki olayları modellemede yetersiz kalmasından dolayı bu tez çalışmasında farklı bir mantıkla “prCell” adını verdiğimiz bir sinyal yolağı programlama dili geliştirdik.

Bu çalışmada Cello programlama dilinin kullandığı Verilog’da kullanılan sınırlı sayıda mantık kapısı yerine, kullanıcının sisteme tanımlayabileceği sınırsız sayıda mantık kapısına(prCell’de mantık kapısı adını kullanmayacağız. Bunun yerine dönüşümü ifade eden “Entity” adını verdiğimiz yapılar kullandık) imkan sağlamış olduk.

Sinyal yolağı çizim uygulamalarında her ne kadar proteinler ve genlere bilgilerine ulaşmak için biyolojik veri tabanına bağlantı olsa da çizim esnasında çok fazla bağlantı kuralları olmadığından dolayı, sinyal yolağı programlama yapılamamaktadır. Buna karşın prCell’de başta kullanıcının tanımladığı kurallarla sinyal yolağı tasarlanırken uyulmak zorunda olunan kurallar mevcuttur. En temel kurallardan bir girdi çıktı eşleşmesini şart koşmasıdır. Girdi çıktı eşleşmesi yoksa kesinlikle bağlantı kurulamaz. Kullanıcı tanımladığı bir etkileşime veya tanımladığı bir entity yapısının bağlantı kurallarına uymak zorundadır. Aksi durumda sistem bağlantıya izin vermeyecektir.

3. GEREÇ VE YÖNTEMLER

3.1. Nesnelere ve Etkileşimler

Daha önce sinyal yollarının, bir hücredeki moleküler yapılar arasındaki belirli bir ürüne veya hücrede bir değişikliğe yol açan bir dizi etkileşimler bütünü olduğunu belirtmiştik. Şimdi de sinyal yolağında var olan moleküler yapıları ve bunlar arasındaki etkileşimleri modellemeye çalışalım.

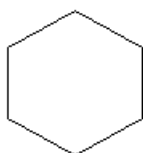
Bu bölümdeki nesne modelleri ve etkileşim modelleri [34] ve [35]'den esinlenerek tasarlanmıştır.[EK1],[EK2]

3.1.1. Sinyal Yolağında Kullanılan Temel Nesnelere

Sinyal yollarının yapısında temelde proteinler, moleküller, protein kompleksleri ve dönüşüm sağlayan yapılar vardır. Bu yapılar birbirleriyle etkileşim oluştururlar.

Bileşikler

Bileşikler protein ve protein kompleksi dışında sinyal yolağı diagramında varolan, küçük molekül yapılarından ve diğer kimyasal maddelerden oluşan biyolojik sistemle ilişkili yapılardır. Bileşikleri sinyal yolağı diagramında Şekil 3.1'de gösterildiği şekilde modelleyebiliriz.

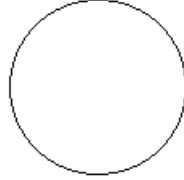


Şekil 3.1 Bileşiklerin model gösterimi

Proteinler

Aminoasitlerin bileşiminden oluşan bu yapılar hücre içindeki birçok fonksiyonda etkin rol oynarlar. Protein sentezi DNA'da ilgili gen tarafından verilen genetik koda göre meydana gelir ve her bir protein hücre içindeki olaylarda görev yapar. Proteinler, hücre içindeki baş aktörlerdir, genlerde kodlanan bilgilerin belirttiği görevleri yerine getirdiği söylenir[7]. Farklı fonksiyon setlerine de izin veren proteinlerin temel özelliği, diğer molekülleri spesifik ve sıkı bir şekilde bağlama yetenekleridir. Protein-protein etkileşimleri ayrıca enzimatik aktiviteyi

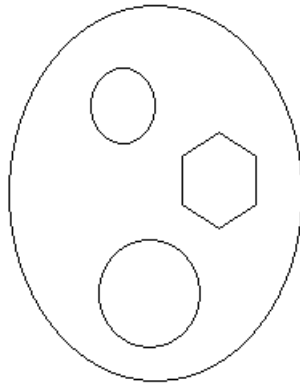
düzenler, hücre döngüsü boyunca ilerlemeyi kontrol eder ve ortak bir biyolojik fonksiyonla yakından ilişkili birçok reaksiyonu gerçekleştiren büyük protein komplekslerinin birleştirilmesine izin verir.[11].Sinyal yolağı tasarımında proteinleri çember şeklinde Şekil 3.2'deki gibi modelleyebiliriz.



Şekil 3.2 Proteinlerin model gösterimi

Kompleks Yapılar

Bir protein kompleksindeki proteinler kovalent olmayan protein-protein etkileşimleriyle bağlanır ve farklı protein kompleksleri zaman içinde farklı stabilite derecelerine sahiptir[11]. Bu kompleksler pek çok biyolojik işlemin temel taşıdır ve birlikte çok çeşitli biyolojik işlevler gerçekleştiren çeşitli moleküler makineler oluştururlar. Protein kompleksi oluşumu bazen kompleks elemanların bir veya daha fazlasının aktifleştirilmesine veya inhibe edilmesine hizmet eder ve bu şekilde, protein kompleksi oluşumu fosforilasyona benzer olabilir[12]. Bireysel proteinler, çeşitli farklı protein komplekslerinin oluşumuna katılabilir. Farklı kompleksler farklı işlevler gerçekleştirir ve aynı kompleks çeşitli faktörlere bağlı olan çok farklı işlevler gerçekleştirebilir. Şekil 3.3'de kompleks modeli çizilmiştir.



Şekil 3.3 Protein kompleks yapılarının modellenmesi

Entity

Hücre içindeki gen ifadelerinin ortaya çıkmasında protein sentezine sebep olan proteinler ile bu transkripsiyonu kontrol eden proteinlerin içinde bulunduğu dönüşümü simgeleyen yapılara denir. Bir proteinin oluşumunda başlangıç görevi yapan proteinler entity yapısı için birer girdidir. Bu olayı olumlu veya olumsuz etkileyen proteinler ve protein kompleksleri de kontrol proteinleri olarak adlandırılır. Bu etkileşimin sonucunda yeni protein ortaya çıkar. Bu proteinler ise entity için birer çıktıdır. Entity yapısı aynı zamanda biyokimyasal reaksiyonları da temsil edebilir. Biyokimyasal reaksiyonları entity ile modellerken girdiler ve çıktılar birer bileşik olabilirken bu reaksiyonun kontrol proteinleri ise enzimlerdir. Her iki durumu da girdi ve çıktı özelliklerine göre entity yapısıyla Şekil 3.4'de gösterildiği gibi modelleyebiliriz.



Şekil 3.4 Entity yapısının model gösterimi

3.1.2. Etkileşimler

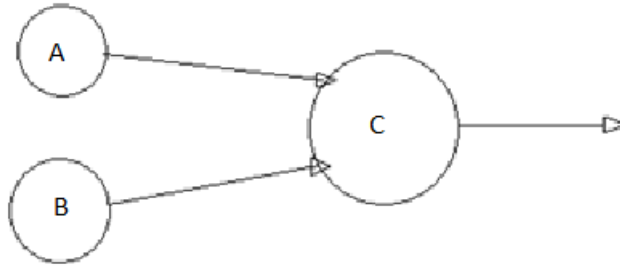
Sinyal yollarında proteinler, moleküller ve diğer kompleksler arasında bir dizi etkileşimler olur. Sinyal yolağı bu etkileşimlerin zincirleme olarak birbirine bağlanmasıyla meydana gelir. Eğer bir sinyal yolağı tasarlayacaksanız bu etkileşim türlerini de modellemeniz gerekmektedir.

Protein-Protein Etkileşimi

Protein-protein etkileşimleri, bir proteinin başka protein veya proteinler sayesinde aktifleşmesi durumunda ortaya çıkar. Mevcut haliyle aktif olamayan proteinleri başka proteinler aktif hale getirir. Sinyal yolağı tasarımında sıkça göstermemiz gereken bu durumu nasıl ifade etmeliyiz?

Protein-protein etkileşimlerini tasarım olarak göstermek istersek aşağıdaki şekille ifade edebiliriz. Proteinlerin birbirleriyle etkileşiminde bir proteinin işlev kazanması veya işlevini değiştirmesi için başka proteinlerin kendisine etki etmesi gerekmektedir. Bu durumda hangi proteinlerin kendisine etki ettiği ve bunun sonucunda işlevinin değiştiğini biliyorsak ilgili proteinleri proteinin input kısmına bağlamış oluruz.

Şekil 3.5’de A ve B girdi proteinleri C proteini ile etkileşim halindedir. A ve B proteinleri C proteinin aktive ya da inhibe ediyor olabilir veya biri aktive ederken diğeri inhibe ediyor da olabilir. Bu durum Şekil 3.5 modellenmiştir.

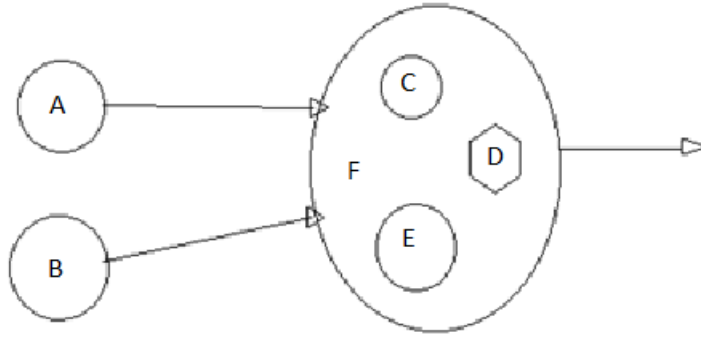


Şekil 3.5 Protein-protein etkileşiminin model gösterimi

Protein-Kompleks Etkileşimi

Kompleks yapılar da tek başlarına aktif olmayabilirler. Onların aktif olabilmesi için bazı proteinlere ihtiyaç duyabilirler. Bu durumda kompleks yapının girdi kısmına bu yapıyı aktifleştirecek proteinleri bağlamamız gerecektir. Bu durumu aşağıdaki şekilde modelleyebiliriz

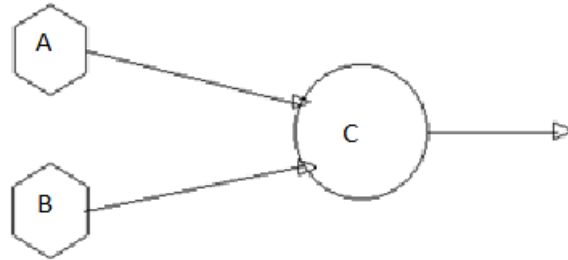
Şekil 3.6’da A ve B proteinleriyle C, E proteinleri ve D molekülünde oluşan kompleks yapıyla etkileşim halindedirler. Burada girdi proteinleri olana A ve B, F kompleks yapısına bir etkileri söz konusudur. Protein kompleksinin içinde bulundurduğu diğer yapılar ve etkileşimde olduğu bileşik ve proteinlerin gösterimi Şekil 3.6’de olduğu gibi tasarlanmıştır.



Şekil 3.6 Protein-kompleks etkileşiminin model gösterimi

Bileşik-Protein Etkileşimi

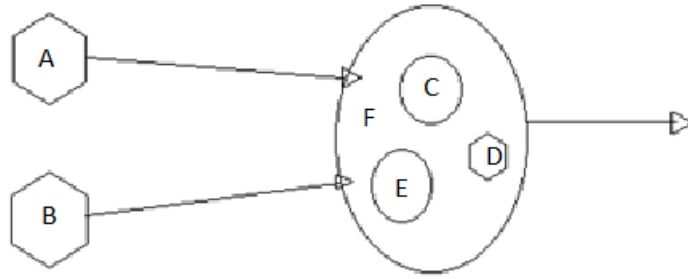
Proteinleri her zaman diğer proteinler aktif hale getirmez. Bazen de bileşikler tarafından aktive edilirler. Sinyal yolağı tasarımında bu durumu aşağıda şekilde gösterebiliriz. Şekil 3.7’de A ve B molekülleri C proteinine etki etmektedir.



Şekil 3.7 Bileşik-protein etkileşiminin model gösterimi

Bileşik-Kompleks Etkileşimi

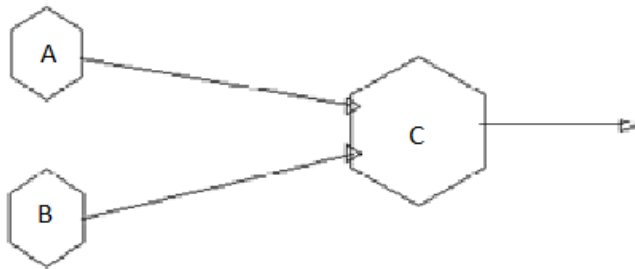
Kompleks yapıların da aktiveleşme durumları bazen molekül yapılar aracılığıyla olabilir. Bu olay aşağıdaki şekilde tasarlanmıştır. Şekil 3.8’de ise A ve B molekülleri F kompleksine etkisi görülmektedir.



Şekil 3.8 Bileşik-protein kompleksi etkileşiminin model gösterimi

Bileşik-Bileşik Etkileşimi

Bir diğer durum da molekül-molekül etkileşimidir. Bir molekülün diğer iki molekül tarafından aktifleştirilmesi gösterilmiştir. Şekil 3.9da A ve B molekülleri C molekülü ile etkileşim kurmaktadır.



Şekil 3.9 Bileşik-bileşik etkileşiminin model gösterimi

3.1.3. Gen İfadesi ve Biyokimyasal Reaksiyonların Modellenmesi

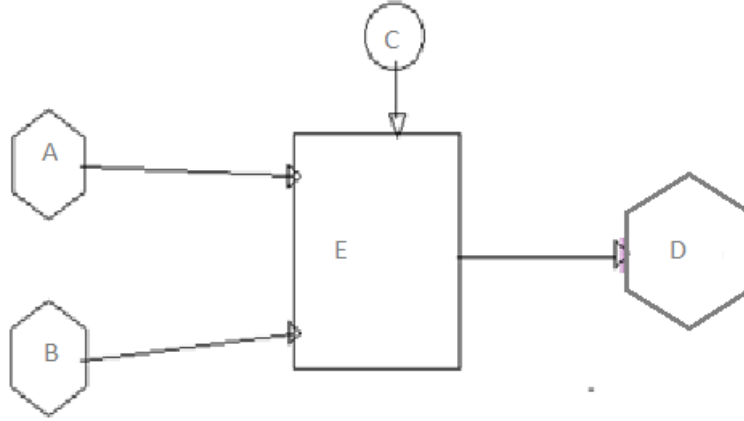
Entity Yapısı ve Etkileşimi

Entity temelde girdiler çıktılar ve kontrol nesnelerinde oluşur ve bir dönüşümü modeller. Proteinlerin, bileşiklerin girdi olabileceği ve yine protein ve protein komplekslerinin kontrolünde başka bileşik ve proteinlerin çıktı olduğu bir dönüşüm modellemesidir. Çıktılar başka bir entity'nin girdileri olacaktır.

Aşağıda gördüğümüz ilk şekilde girdiler bileşik çıktı da bir bileşik. Entity'nin üst kısmındaki proteinin kontrolünde bir biyokimyasal dönüşümün modellendiğini görüyoruz.

-Biyokimyasal reaksiyonlar:

Aşağıdaki Şekil 3.10da girdideki A ve B molekülleri/bileşikleri C proteininin(enziminin) kontrolünde D molekülüne/bileşiğine dönüşmektedir.

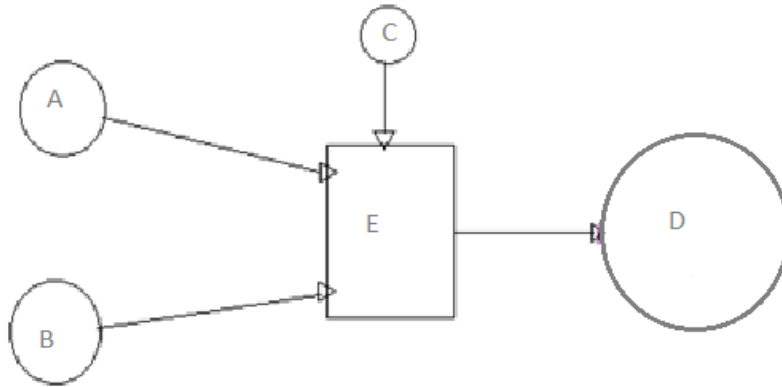


Şekil 3.10 Biyokimyasal reaksiyonların model gösterimi

Şimdi de gen ifadesinin nasıl modellendiğine bir göz atalım. Aşağıda gördüğünüz şekilde girdi kısmında iki protein var. Bu proteinler çıktıda oluşan proteinin oluşumunun başlamasına sebep olan proteinler. Entity'nin üst kısmında gördüğünüz protein ise oluşan gen ifadesinin aktive eden ya da inhibe eden kontrol proteinleridir.

-Gen ifadesinin gösterimi:

Şekil 3.11 de ise A ve B proteinleri C proteininin kontrolünde D proteinin oluşmasını sağlamaktadırlar. Burada D proteini bir gen ifadesi olarak karşımıza çıkmaktadır.



Şekil 3.11 Gen ifadesinin model gösterimi

3.1.4. Bağlanma

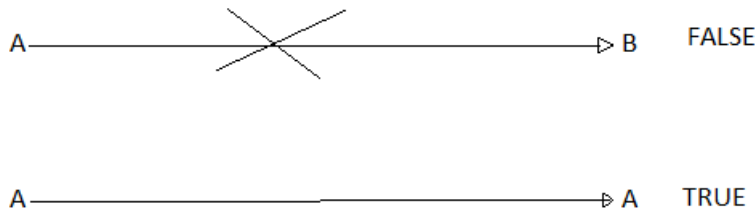
Yapay sinyal yolağı çiziminde nesnelerin birbirleriyle olan etkileşimi bağlanmayla gösteririz

Bağlanma Kuralları

Diagram içerisindeki nesnelerin sinyal yolağının akış yönüne doğru etkileşimde olması gerekir. Soldan sağa doğru sinyal yolağının iletimi gösterilmelidir. Bağlanma için girdi çıktı eşleşmesi şarttır.

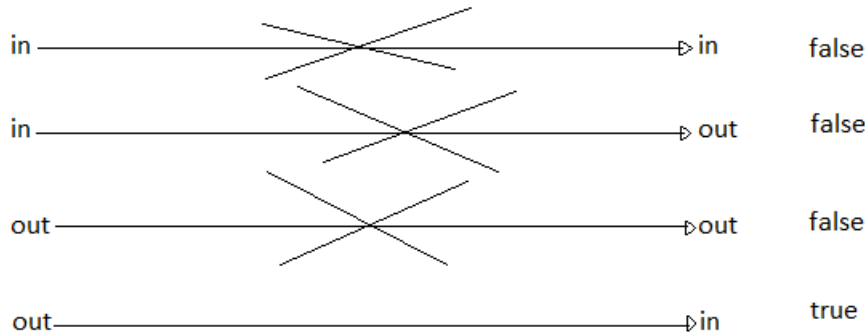
Girdi Çıktı Eşleşmesi

Sinyal yolağı tasarımında bir nesnenin sadece çıktısı bir başka nesnenin girdisine bağlanabilir. Bir nesnenin girdisinden bağlantı başlatılamaz. Başlangıç noktası her zaman çıktı olması gerekir. Eğer başlangıç noktası çıktı olan, bağlanma noktası diğer nesnenin girdisi olan bir durumda bağlantının olabilmesi için girdi çıktı eşleşmesi olması şarttır. Girdi çıktı eşleşmesi bir nesnenin çıktısı ile başka bir nesnenin girdisinin aynı isimde olması durumunda gerçekleşir. Çıktı isim olarak ilgili nesnenin adıyla aynıdır. Bu eşleşme kuralı Şekil 3.12’de gösterilmiştir.



Şekil 3.12 Eşleşme kuralının gösterimi

Yukarıdaki şekilde görüldüğü üzere başlangıç ve bitiş noktaları aynı ise bağlanma gerçekleşiyor, aynı değilse bağlantı gerçekleşmiyor. Sinyal yolağı tasarımındaki en temel kurallardan biri olan girdi çıktı eşleşme şartı tam olarak aşağıda gösterildiği gibidir. Girdi-çıkı kuralının modelini Şekil 3.13’de gösterilmiştir.



Şekil 3.13 Girdi-çıkı kuralının gösterimi

3.2. İş Akışı

3.2.1. Üyelik

Yapay sinyal yolağı sistemi web tabanlı olacağı için her kullanıcı sisteme üye olmak zorundadır. Kendilerine ait kullanıcı adı ve şifrelerle sisteme gireceklerdir.

3.2.2. Temel Nesnelerin Kaydedilmesi

Yapay sinyal yolağı tasarlayabilmek için sisteme bileşik protein protein kompleksleri ve entity gibi temel nesnelere tanımlamamız gerektir. Sisteme ne kadar nesne kaydederse sistem o derece yetenekli hale gelecektir.

Molekül Kaydı

Bileşik kaydı bileşiğin adı, sembolü ve ne işe yaradığı gibi temel bilgilerinin kaydedilmesi gerekmektedir.

Protein Kaydı

Protein kaydı proteinin adı, sembolü ve ne işe yaradığı gibi temel bilgilerinin kaydedilmesi gerekmektedir.

Kompleks Yapı Kaydı

Kompleks yapı kaydı kompleksin adı, sembolü ve ne işe yaradığı gibi temel bilgilerinin kaydedilmesi gerekmektedir.

Entity Kaydı

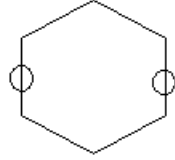
Entity yapı kaydı entity'nin adı, sembolü ve ne işe yaradığı gibi temel bilgilerinin kaydedilmesi gerekmektedir

3.2.3. Nesne ve Entity Yapısına Port Eklenmesi

Port eklemek nesne ve entity'lerin etkileşimde olduğu diğer nesnelere bağlantılarını yapabilmek için gereklidir.

Moleküllere Port Eklenmesi

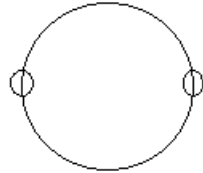
Bir moleküle port eklemek aslında onu etkileyen başka bir hücrenel yapıyı belirlemek demektir. Girdi portları ilgili molekülü etkileyen yapıları belirlemek için kullanılırken çıktı portu tekdir ve ilgili molekülü sembolünü alır. Bileşiğin port eklenmiş hali Şekil 3.14’de gösterilmiştir.



Şekil 3.14 Bileşiklere port eklenmesinin model gösterimi

Proteinlere Port Eklenmesi

Proteinler başka bir proteini bazen aktive edebilir inhibe edebilir baskılayabilir ya da durumunu değiştirebilir. Bunu diagramda gösterebilmek için bir proteine port eklerken yani onu etkileyen proteini belirlerken aynı zamanda nasıl bir etki yarattığını da belirtmemiz gerekmektedir (Şekil 3.15). Bu etkinin çeşidi eklenen portun rengi ile simgeleyebiliriz.

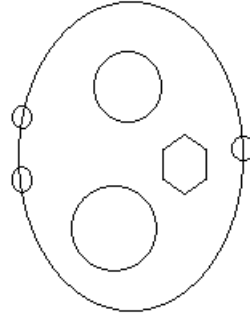


Şekil 3.15 Proteinlere port eklenmesinin model gösterimi

Kompleks Yapılara Port Eklenmesi

Kompleks yapı kaydedebilmek için öncelikle bu yapıda bulunacak nesnelere daha önce sisteme kaydının yapılmış olması gerekmektedir. Kompleks yapılar da tıpkı proteinler gibi başka proteinler veya moleküller tarafından aktive etmek inhibe etmek bir etkiye maruz kalabilir. Aşağıdaki şekilde içinde başka proteinleri barındıran bir kompleks yapı bulunmaktadır. İçinde barındırdığı proteinler de portlarıyla birlikte temsil edildiği görülmektedir (Şekil 3.16). Kompleks yapının girdi bölümünde iki adet port gösterilmiştir. Bu

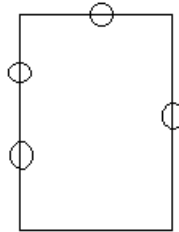
portların sembolü ve portların renkleri eklendiğinde bu yapıyı hangi molekül veya proteinin nasıl etkilediğini görmüş olacağız.



Şekil 3.16 Protein komplekslerine port eklenmesinin model gösterimi

Entity Yapılarına Port Eklenmesi

Entity yapısı tam olarak bir dönüşümü simgeleyecektir. Bu dönüşüm bir biyokimyasal reaksiyon alabileceği gibi bir genin ifadesi de olabilir. Diğer nesnelere aksine bu dönüşümü kontrol edecek olan proteinleri simgeleyen kontrol portları da mevcuttur. Bu portlar entity'nin üst kısmında bulunur. Girdi, çıktı portlarına ek olarak kontrol portları da entity üzerinde gösterilir (Şekil 3.17). Kontrol portlarının etkisi dönüşümü artırıcı veya baskılayıcı olabilir ve bu durumlar yeşil ve kırmızı renkle gösterilebilir.



Şekil 3.17 Entity yapılarına port eklenmesinin model gösterimi

3.2.4. Proje Oluşturma ve Diagram Çizme

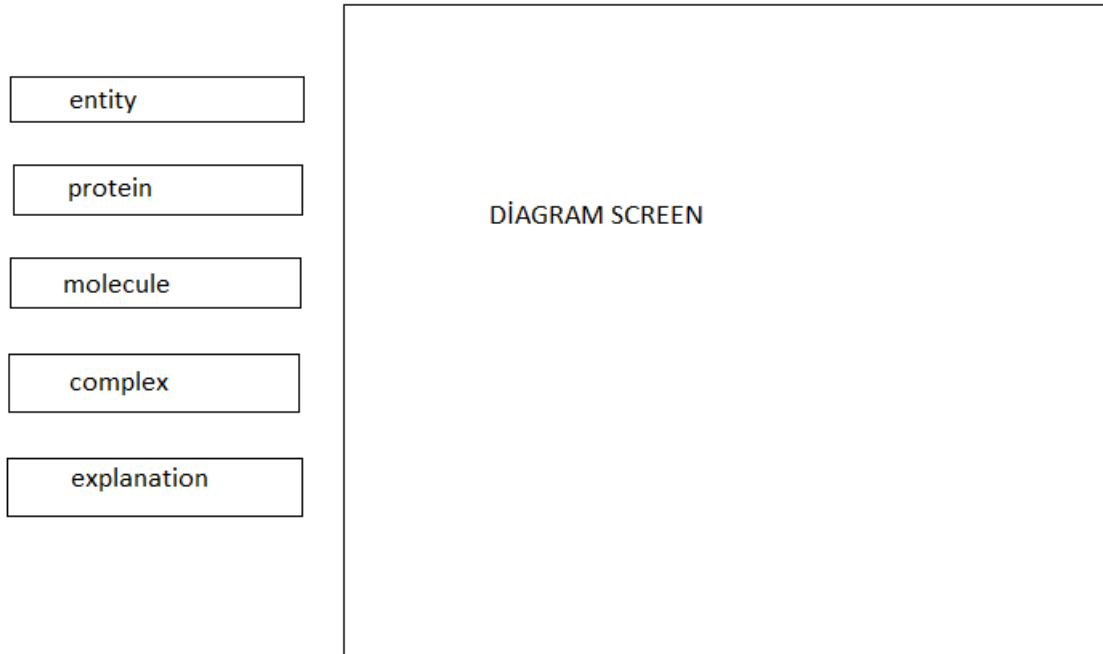
Temel nesnelere ilgili veri kayıtlarını bitirdikten sonra sinyal yolağı tasarlayabiliriz. Bunun için önce proje kaydı yapmamız gereklidir

Proje Kaydı

Sisteme tasarlayacağımız sinyal yolağının adını taşıyan bir proje kaydı yapmamız gerekir. Proje kaydı temel nesnelerin kaydında olduğu gibi adı sembolü ve kısa bir açıklamayı içerir. Her bir sinyal yolağı tasarımı için bir proje kaydı yapmak gerekir.

Diagram Ortamına Nesne Ekleme

Daha önce oluşturduğumuz projeyi açtığımızda tasarım yapacağımız alanın yan tarafında bir menü olmalı ve bu menüden protein bileşik protein kompleksi ve entity ekleyebilmeliyiz. Şekil 3.18’de bu ekranı yansıtmaya çalıştık.



Şekil 3.18 Diagram çizim ekranının model gösterimi

3.3. Yazılımın Teknik Özellikleri

Yapay sinyal yolağı tasarlayabileceğimiz yazılımın adı “prCell” olacak. Web tabanlı olacak olan bu yazılım internette www.prcell.net alan adı ile ileride yayınlanacaktır. Bu yazılım yazılırken kullanılan araçlar aşağıda açıklanmıştır.

3.3.1. Veri tabanı dili

Bu yazılım projesinde veri tabanı olarak “MSSQL” veri tabanı kullanılmıştır. Microsoft ürünü olan msSQL’in 2014 sürümü uygun görülmüştür. Veri tabanı sistematik ve düzenlenmiş

verilerin biraraya getirildiği ve kaydının tutulduğu bir yapıdır. Daha geniş bir ifadeyle birbirleri ile bir şekilde ilişkili olan verilerin farklı amaçlar için tutulmasına, gerektiğinde işlenmesine ve değiştirilmesine olanak sağlayan yazılımlara veri tabanı denir.[36]

MSSQL Microsoft firmasının geliştirdiği bir veri tabanı türüdür. Yazılım yapmak istediğimizde, iş analizinden sonra, kullanacağımız yazılım diline de uygun olmak koşuluyla bir veri tabanı sistemi seçmemiz gereklidir. Bu çalışmamızda programlama dili olarak .NET MVC kullanacağımızdan dolayı uyumlu olaması için veri tabanı dili olarak msSQL kullanılmıştır.

3.3.2. Programlama dili

Visual studio, Microsoft tarafından üretilen ve grafik kullanıcı ara yüzleri, windows formları, web servisleri ya da web uygulamaları geliştirmek için kullanılan bir tümleşik geliştirme ortamıdır. Visual Studio uygulaması içerisinde sadece Microsoft Windows tarafından desteklenen kodlar kullanılmaktadır.

Visual Studio'nun yazılım üretmek için bize bir geliştirme ve derleme ortamı sunan bir büyük yazılım olduğunu söyleyebiliriz. Visual Studio yazılımları yazılım mühendislerine ileri seviye özelliklerinin verdiği imkanlar sayesinde daha kısa süre içerisinde uygulamalarını geliştirme fırsatı sunar. Visual Studio yazılımını kullanarak nesneye yönelik yazılımları, internet uygulamalarını ve web servislerini hızlı ve kolay bir şekilde derleyebilir ve geliştirebilirsiniz.

3.3.3. Yazılım Mimarisi

Yazılım mimarisi olarak MVC kullandığımız bu teknolojinin nasıl bir mantığa dayandığına baktığımızda, MVC(model, view, controller) üç yapıdan oluştuğunu görürüz. Model bileşeni veriyi temsil eder ve bu kısmı veri tabanındaki tabloları bir model olarak alarak sağlar. Yazılımın veri ile ilişkilerinin bu katman sağlar. Verilerin veri tabanına işlenmesi model aracılığıyla olur.[37]

View model model'lerin kullanıcı kısmında görselleştirilmesiyle ilgilenir. Modelde tablodaki bir alanı sayıyla tutabilirsiniz ancak kullanıcıya sayıyı yerine bunun karşılığını gösterirsiniz. Örneğin bir tabloda iller alanına modelde plaka numaralarını tutarken view modelde bunlara karşılık gelen illerin gerçek isimlerini gösterirsiniz. View model modelin kullanıcıları hitap eden kısımdır.

Controller içinde barındırdığı fonksiyonlar sayesinde, yazılımın işleyişini yönetir ve denetler, view ile model arasındaki koordinasyonu sağlar. Controller kullanıcıdan view aracılığıyla girdiyi alır, sonra modeli de kullanarak kullanıcının talebini gerçekleştirmek üzere bazı işlemler yapar ve sonucu view'e tekrar gönderir. Controller, verinin nasıl elde edildiğiyle ya da yaptığı işlemler sonucu ortaya çıkan sonucun kullanıcıya nasıl iletileceğiyle ilgilenmez. Controller arka planda kullanıcının görmediği bütün işlemleri yapan ve işlenen veriyi view bileşenine aktararak kullanıcının isteklerini yerine getiren katmandır.[38] Controllerın başlıca görevleri şunlardır:

- Model ve View arasındaki bilgi alışverişini sağlamakla görevlidir.
- Kullanıcıların View aracılığıyla yaptıkları işlem taleplerini Model'e aktarır, Model'den elde ettiği (veri tabanından elde ettiği) de View aracılığıyla kullanıcıya gönderir.
- MVC mimarisinde temel mantık Model ve View yapılarının ayrılmasıdır. Bu iki katman arasındaki iletişimi sağlayan mimari katmana Controller denir.

Asp.NET MVC yeni bir yazılım mimarisi olduğundan dolayı kodlamayı sistemli ve sade hale getirmiştir.

3.3.4.Javascript Kütüphanesi

Yukarıda oluşturduğumuz modellerin yazılımda da oluşturulabilmesi için .net mvc'ye bununla ilgili bir kütüphane eklememiz yazılımın amacı için en önemli şartlardan biridir. Bu amaçla yapılan araştırmada en uygun kütüphanenin JointJS olduğu anlaşılmıştır. prCell uygulamasında oluşturulan modeller JointJS kullanılarak oluşturulmuştur[38]. Bu kütüphaneyi tercih etmemizin nedenleri aşağıda açıklanmıştır.

JointJS, statik diyagramlar veya etkileşimli diyagram araçları oluşturmak için kullanılabilir. Bu uygulamada ücretli ürünü olan RAPPID kullanılmamıştır. JointJS kütüphanesinin ücretsiz, açık kaynak kodları kullanılmıştır

JointJS özelliklerinden bazıları şunlardır:

- . Temel diyagram şekilleri (dikdörtgen, daire, elips, yazı, resim, yol) çizilebilmesi[39]
- . Bilinen diyagramların kullanıma hazır çizim elemanları. Ancak bunların hiçbiri yazılımda kullanılmadı.

- SVG'ye dayalı veya programlı olarak oluşturulan özel şekiller oluşturabilirsiniz.

- Etkileşimli nesnelere ve bağlantılar oluşturabilirsiniz.[40]
- Bağlantı elemanları ile bağlantı şemaları[41]
- Şekillere portlar ekleyebilirsiniz.[42]
- Bağlantı noktalarını temel olarak her yere yerleştirilebilir.
- JSON formatına seri hale getirme / seri kaldırma. Bu özellik şekillerin veri tabanına kaydedilmesini sağlamaktadır.
- Yakınlaştır / uzaklaştır özelliği

3. 4. Veri tabanı Tasarlama

İş analizinden sonra yapılması gereken ilk iş veri tabanı tasarımıdır. Hangi bilgilerin tutulacağını ve bu bilgiler için tasarlanan tabloların birbirleriyle nasıl bir ilişkide olacağını veri tabanı tasarım esnasında göstermemiz gereklidir. Bu uygulama için 50'ye yakın tablo oluşturduk. Bu tabloların bazıları doğrudan veri tutmak yerine tabloların birbirleriyle olan ilişkileri göstermek için oluşturduk. Bazı tablolarda geçici veri tuttuğundan dolayı bu bölümde size sadece temel verilerle ilgili tabloları ve bu tablolarda hangi verilerin nasıl tutulduğunu göstermek istiyoruz.

3.4.1. Temel Nesnelere İlgili Tablolar

Tasarladığımız veri tabanında temel nesnelere ilgili olarak bileşik tablosu(Şekil 3.19), protein tablosu(Şekil 3.20), protein modifikasyon tür tablosu (Şekil 3.21), protein kompleks tablosu (Şekil 3.22) ve protein kompleks yapılarının bileşen tablosu (Şekil 3.23) vardır. Bunların her birinde nesnenin adı sembolü ve açıklaması tutulur. Aşağıda bileşikler için tasarlanan tablonun ekran görüntüsü vardır.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	CompSymbol	varchar(50)	<input type="checkbox"/>
	CompName	varchar(50)	<input type="checkbox"/>
	Exp	varchar(1000)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.19 Bileşik/molekül tablosu

Bileşikler tablosunun alanları "ID", "Userid", "CompName", "CompSymbol" ve "Exp" alanlarından oluşmaktadır. "ID" alanı her bir kaydın tekrarsız olması için "uniqueidentifier" olarak tanımlanmıştır. "Uniqueidentifier" ilgili alanın benzerinin tabloya kaydedilmemesi için tanımlanır. "Userid" alanının olmasının nedeni tabloların kullanıcıya özel olmasından kaynaklanır. Bir kullanıcının kayıtlarının başka kullanıcılar tarafından görülmemesi ve kullanılmaması için "Userid" alanı eklenmiştir. "CompName" bileşiklerin adını tutar. "CompSymbol" ise bileşiklerin sembolünü kaydetmek için eklenmiştir. Son olarak "Exp" alanı bu kayıtla ilgili tanım veya açıklama için ayrılmıştır.

Aşağıda bir diğer temel nesne olan proteinlere ait tablo vardır.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	ProName	nvarchar(500)	<input type="checkbox"/>
	ProSymbol	nvarchar(50)	<input type="checkbox"/>
	Exp	nchar(1000)	<input checked="" type="checkbox"/>
	Modify	nchar(10)	<input checked="" type="checkbox"/>
	ModifiedWith	nchar(10)	<input checked="" type="checkbox"/>
	sProSymbol	varchar(50)	<input checked="" type="checkbox"/>
▶	tProSymbol	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Şekil 3.20 Protein tablosu

Protein tablosunun alanları "ID", "Userid", "ProName", "ProSymbol", "Exp", "Modify", "ModifiedWith", "sProSymbol" ve "tProSymbol" alanlarından oluşmaktadır "ID" alanı her bir kaydın tekrarsız olması için "uniqueidentifier" olarak tanımlanmıştır. "Uniqueidentifier" ilgili alanın benzerinin tabloya kaydedilmemesi için tanımlanır. "Userid" alanının olmasının nedeni tabloların kullanıcıya özel olmasından kaynaklanır. Bir kullanıcının kayıtlarının başka kullanıcılar tarafından görülmemesi ve kullanılmaması için "Userid" alanı eklenmiştir. "ProName" proteinin adını tutar. "ProSymbol" ise proteinin sembolünü kaydetmek için eklenmiştir. "Exp" alanı bu kayıtla ilgili tanım veya açıklama için ayrılmıştır.

Burada eğer protein modifiye olmuşsa "Modify", "ModifiedWith", "sProSymbol" ve "tProSymbol" alanlarının doldurulması gerekir. "Modify" alanı sadece modifiye olup olmadığını, "ModifiedWith" alanı eğer protein modifiye olmuşsa hangi molekülle modifiye olduğunu, "sProSymbol" hangi proteinin modifiye sebep verdiğini, "tProSymbol" de modifiye olan proteini temsil etmektedir. Bir protein modifiye olduğu zaman artık sistem için farklı bir

protein gibi kayda düşer. Sembol olarak da küçük bir değişime uğrayarak sembolünü modifiye olduğu molekülün simgesini ekler. Aşağıda proteinlerin nelerle modifiye olabileceğinin tablosu eklenmiştir. Bu tablo protein tablosundaki "ModifiedWith" alanı ile bağlantılıdır.

	ID	Symbol	Name
▶	1	+p	phosphorylation
	2	-p	dephosphorylation
	3	+g	glycosylation
	4	+u	ubiquitination
	5	+m	methylation
*	NULL	NULL	NULL

Şekil 3.21 Protein modifikasyon tür tablosu

Bir diğer temel nesnemiz olan proteine komplekslerine bakalım. Protein kompleksleri içinde molekülleri ve proteinleri barındıran karmaşık yapılardır ancak burada sadece temel bilgilerini kaydediyoruz.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	CxName	nvarchar(500)	<input checked="" type="checkbox"/>
	CxSymbol	nvarchar(50)	<input checked="" type="checkbox"/>
	Exp	nchar(1000)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.22 Protein kompleks tablosu

Protein kompleks tablosunun alanları "ID", "Userid", "CxName", "CxSymbol" ve "Exp" alanlarından oluşmaktadır. "ID" alanı her bir kaydın tekrarsız olması için "uniqueidentifier" olarak tanımlanmıştır. "Uniqueidentifier" ilgili alanın benzerinin tabloya kaydedilmemesi için tanımlanır. "Userid" alanının olmasının nedeni tabloların kullanıcıya özel olmasından kaynaklanır. Bir kullanıcının kayıtlarının başka kullanıcılar tarafından görülmemesi ve kullanılmaması için "Userid" alanı eklenmiştir. "CxName" protein kompleksinin adını tutar. "CxSymbol" ise protein kompleksinin sembolünü kaydetmek için eklenmiştir. Son olarak "Exp" alanı bu kayıtla ilgili tanım veya açıklama için ayrılmıştır.

Şimdi de bu protein komplekslerinin hangi yapıların birleşiminden oluştuğunu tutan tabloya bakalım.

Kompleks yapıları temel bilgileriyle kaydettikten sonra bu yapıların hangi nesnelere oluşturulduğunu belirlememiz gerekir. Örneğin bir kompleks yapı 3 protein ve 1 bileşiğin birleşiminden oluşabilir. Bu kaydı aşağıdaki tabloda tutarız.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	CplxID	uniqueidentifier	<input type="checkbox"/>
	ProID	uniqueidentifier	<input checked="" type="checkbox"/>
	CompID	uniqueidentifier	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.23 Protein kompleks yapılarının bileşen tablosu

Burada "ID" ve "Userid" ne olduğunu önceden belirtmiştik."CplxID" Kompleks tablosuna bağlı olarak hangi kompleks olduğunu belirtirken, "ProID" bu kompleksin barındırdığı proteinleri tutar."CompID" ise bu yapının içerdiği bileşik veya bileşikler tutmaktadır. Böylece bu tablodan kompleks yapıların nelerden oluştuğunu görebiliriz.

3.4.2. Nesnelere Etkileşim Tabloları

Nesnelerin etkileşim tablosu olarak etkileyen nesne, etkilenen nesne ve etki düzeyi tutulur. Tutulan etkileşim formları (Şekil 3.24) şunlardır:

	ID	Effect
▶	1	Activation
	2	Inhibition
	3	Expression
	4	Repression
	5	IndirectEffect
	6	StateChange
*	NULL	NULL

Şekil 3.24 Etkileşim türleri tablosu

Bir nesne başka bir nesneyi aktive edebilir ya da inhibe edebilir. Bu bilgi etkileşim tablolarında "EffectID" olarak tutulur. Bu "id" yukarıdaki tabloyla ilişkili olduğundan "id"ye

karşılık gelen etkinin oluştuğunu anlarız. Aşağıda proptein-protein etkileşim tablosunu görüyorsunuz (Şekil 3.25).

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	sourceProID	uniqueidentifier	<input checked="" type="checkbox"/>
	targetProID	uniqueidentifier	<input checked="" type="checkbox"/>
	EffectID	nvarchar(50)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.25 protein-protein etkileşim tablosu

Her tabloda olduğu gibi burada da “ID” ve “Userid” standart olarak karşımıza çıkıyor. “sourceProID” etkileşimde “etkileye protein” olurken “targetProID” ise “etkilenen protein” olarak kabul edilir. “EffectID” ise etki şeklini belirleyen alan olarak tabloya eklenmiştir.

Sadece protein-protein etkileşimleri değil farklı etkileşim türleri de vardır. Bunlarda biri de bileşik-kompleks yapı etkileşimidir. Burada etkileyen taraf “bileşik” etkilenen taraf ise “kompleks” yapıdır. Aşağıdaki ekran görüntüsünde (Şekil 3.26) bu durumun veri tabanında nasıl tutulduğunu göstermektedir.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	sourseComp	uniqueidentifier	<input type="checkbox"/>
	targetCx	uniqueidentifier	<input type="checkbox"/>
	EffectID	nvarchar(50)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.26 Bileşik-protein kompleks etkileşim tablosu

Yukarıda gördüğünüz ekran görüntüsünde “sourceComp” kompleks yapıyı etkileyen bileşik, “targetCx” ise etkilene kompleks yapıyı temsil eder. “EffectID” bu etkinin nasıl olduğunu bize gösterir.

Şuana kadar protein-protein ve bileşik-kompleks etkileşim tablolarının özelliklerini inceledik. Benzer şekilde protein-kompleks, protein-bileşik, kompleks-protein, kompleks-kompleks, kompleks-bileşik, bileşik-protein ve bileşik-bileşik tabloları da veri tabanında yer

almaktadır. Tabloların tasarımları yukarıdaki etkileşim tablolarıyla benzer olduğu için ekran görüntülerini buraya eklemedik.

3.4.3. Entitiy Tabloları

Temel nesne olarak entitiy kaydı yapıldıktan sonra entity için giriş çıkış ve kontrol portları eklemek gerekecektir. Port da protein bileşik gibi bir nesneyi temsil eder ve entity'e port ekleyerek entity'nin etkileşim halinde olduğu nesnelere belirtmiş oluruz. Bununla ilgili tablolar şunlardır:

- entity-input
- entity-output
- entity-protein
- entity-kompleks

Entity-protein ve entity-kompleks tabloları entitynin kontrol mekanizmasını temsil eder. Yani entity bizim için bir dönüşümü modellediği için bu dönüşümü proteinler ve kompleks yapılarla kontrol ederiz. Entity-input ve entity-output tabloları bu dönüşümde girdi ve çıktılarını temsil edecektir. Entity-input ve entity-output tablolarında bileşik protein ve komplekslerden herhangi birilerini ekleyebiliriz. Bir entity yapısında girdi olarak bir protein kaydı ve çıktı olarak da başka bir protein kaydı varsa ilk proteinin ikinci proteinin sentezlenmesinde başlangıç etkisi olduğu anlaşılır. Kontrol proteinleri ise sadece dönüşümü olumlu veya olumsuz etkileyebilir. Girdi eğer bir bileşik ve çıktı da başka bir bileşikse entity yapısı bir biyokimyasal dönüşümü temsil etmiş olur.

Aşağıdaki tabloların veri tabanında neden tutulduğunu daha iyi anlamak için Şekil 3.10 ve Şekil 3.11 model gösterimlerini göz önünde bulundurmanız gerekmektedir. Aşağıda gösterilmiştir:

- entity tablosu
- entity-bileşik input tablosu
- entity-protein input tablosu
- entity-protein output tablosu ve
- entity-protein kontrol tablosu

Benzer mantık diğer tablolarda olduğu için –entity-protein kompleksi kontrol tablosu ve –entity-bileşik output tablosu eklemeye gerek duyulmamıştır.

Gen ifadesi ve biyokimyasal reaksiyonları temsil eden entity yapısı aslında tam olarak bir dönüşümü modeller. Bu dönüşümde girdiler, çıktılar ve kontrol proteinleri vardır. Aşağıdaki tablolarda (Şekil 3.27 - Şekil 3.31) entity yapısının verilerinin nasıl tutulduğunu göreceğiz.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	EntName	nvarchar(500)	<input checked="" type="checkbox"/>
▶	EntSymbol	nvarchar(50)	<input checked="" type="checkbox"/>
	Exp	nchar(1000)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Şekil 3.27 Entity tablosu

Yukarıdaki tabloda tasarım elemanlarından biri olan entity yapısının temel verilerinin hangi alanlarla tutulduğunu görüyoruz. "ID" ve "Userid" bütün tablolarda var olan alanlardır."EntName" entitiy'nin adını "EntSymbol" entitiy'nin sembolünü ve "Exp" ise diğer tablolarda olduğu gibi açıklamayı tutmaktadır. Bu tablo temel nesnelerin tabloları gibi temel verileri tutmaktadır. Entity kaydı temel verileriyle kaydedildikten sonra bu entity yapısının girdi, çıktı ve kontrol yapılarının da kaydedilmesi gerekir.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	EntID	uniqueidentifier	<input type="checkbox"/>
	InputCompID	uniqueidentifier	<input type="checkbox"/>
	PropertyID	nchar(100)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.28 Entity-bileşik input tablosu

İlk olarak entity'nin girdi kaydını tutulmasına bakalım. Girdi olarak entity için bileşik olabileceği gibi protein ve kompleksler de girdi olabilir. İlk tablomuz input bileşik tablosu. Yani entity'nin girdisi bir bileşik ise bilgileri bu tabloda tutulacak. Yukarıda gördüğümüz bu tabloda ""EntID" entity bilgisini tutarken "InputCompID" de bu entity'nin girdisi olacak bileşiği tutmaktadır."PropertyID" ise bu girdinin entity'yi nasıl etkilediğini tutmaktadır.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	EntID	uniqueidentifier	<input type="checkbox"/>
	InputProID	uniqueidentifier	<input type="checkbox"/>
	PropertyID	nchar(100)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.29 Entity-protein input tablosu

İkinci tabloda değişen alan "InputProID" alanıdır, diğer alanlar aynıdır ve herhangi değişiklik yoktur. Bu tablo entity'inin girdisi bir proteince ilgili kayıtları tutmaktadır. Genellikle gen ifadesi oluşumlarında gen ifadesinin başlamasına sebep olan proteinleri bir girdi olarak burada gösterebilirsiniz. Bunların dışında girdi olarak kompleks yapıların kaydını tutan ve benzer alanlara sahip tablo da mevcuttur.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	EntID	uniqueidentifier	<input type="checkbox"/>
	OutputProID	uniqueidentifier	<input type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.30 Entity-protein output tablosu

Yukarıda gördüğünüz tablo da entity yapısında çıktı proteinleri tutmaktadır."OutputProID" entity dönüşümünden sonra oluşacak proteinin kaydını tutmamızı sağlar. Çıktı olarak bileşikleri de sisteme başka bir tabloyla dahil edebiliriz. Bunun kaydı için de "OutputProID" yerine "OutputCompID" tanımlarız.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	EntID	uniqueidentifier	<input type="checkbox"/>
	ControlProID	uniqueidentifier	<input type="checkbox"/>
	EffectID	nvarchar(50)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.31 Entity-protein kontrol tablosu

Şimdiki tabloda ise entity dönüşümünü kontrol eden proteinlerin kaydını tutan tabloya bakalım. Yukarıda gördüğünüz bu taloda “ControlProID” alanı ilgili entity’i kontrol eden proteinleri gösterir.”EffectID” ise bu kontroldeki etkisinin ne olacağı ile ilgili bize bilgi verir.

3.4.4. Geçici ve Gerçek tablolar

Veri tabanında hemen hemen bütün tabloların geçici ve gerçek hali vardır. Bir entity tanımlarken entitynin girdilerini, çıktılarını ve kontrol yapılarını eklerken veya çıkarma ve düzenleme yaparken veriler geçici tablolarda tutulur. Entity yapısının son halini verdikten sonra kesin olarak kaydettiğimizde ise geçici tablolardaki bilgiler gerçek tablolara aktarılır ve geçici tablolardaki veriler silinir. Kesin kayıt yapmadığınız sürece entity yapısını düzenleyebilirsiniz.

3.4.5. Diagram Tablosu

Yapay sinyal yollarını tasarladıktan sonra bunları sisteme kaydederiz. Mevcut tasarım şekillerden ibarettir. Bu tasarım json formatında diagram tablosuna kaydedilir.(Şekil 3.32’de görüldüğü üzere) Bu çok küçük bir alan kapladığından çok büyük ve karmaşık sinyal yollarını tasarlamak sistem için bir problem olmayacaktır. Aynı şekilde daha önce yapılan tasarımlarda mevcut tablodan çekilerek ara yüze taşınacak ve üzerinde ekleme ve düzenleme yapıldıktan sonra tekrar diagram tablosuna son haliyle kaydedilecektir.

	Column Name	Data Type	Allow Nulls
🔑	ID	uniqueidentifier	<input type="checkbox"/>
	Userid	uniqueidentifier	<input type="checkbox"/>
	ProjSymbol	uniqueidentifier	<input type="checkbox"/>
	ProjName	uniqueidentifier	<input type="checkbox"/>
	Cnvs	nvarchar(50)	<input checked="" type="checkbox"/>
	Exp	nvarchar(500)	<input checked="" type="checkbox"/>
	Date	datetime	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

Şekil 3.32 Diagram tablosu

3.5. Nesnelerin Modellenmesi

3.5.1. Temel Nesnelerin Diagramda Gösterim Şekilleri

Daha önce yaptığımız analiz çalışmasında oluşturduğumuz modellerin prCell yazılımında kullanılacak gerçek modellerini oluşturmamız gerekir. Bu bölümde, sinyal yolağı tasarımı için kullanılacak bileşik, protein, kompleks yapılar ve entitylerin gerçek modellerini inceleyeceğiz.

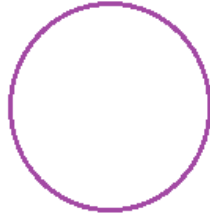
Moleküllerin Modellenmesi

Sinyal yolağı tasarlanırken temel nesnelere bazı şekillerle modelleriz. Bunların içinde molekül yapıları (compound) altıgen şeklinde gösteririz. (Şekil 3.33'de modellendiği gibi)



Şekil 3.33 prCell yazılımında bileşik/molekül gösterimi

Proteinlerin Modellenmesi

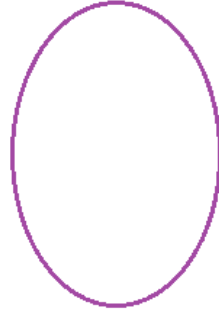


Şekil 3.34 prCell yazılımında protein gösterimi

Proteinler ise mor renkli çember şeklinde diagramda Şekil 3.34'deki gibi gösterilir. Eğer modifiye olmuş bir proteince rengi normal proteine göre daha koyu yapılmıştır.

Kompleks Yapıların Modellenmesi

Proteinler kompleksleri ise içinde barındırdığı protein ve molekülleri barındıran oval bir yapıyla temsil edilir. (Bknz. Şekil 3.35) Bu oval yapı hareket ettirildiğinde içindeki bileşenlerle birlikte hareket eder.



Şekil 3.35 prCell yazılımında protein kompleksi gösterimi

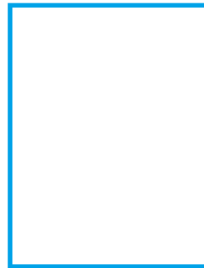
Nesnelerin Birbirlerine Bağlanması

Diagramda kullanılan nesnelerin birbirleriyle bağlanması sinyal yolağının yönünü ve etkileşimleri gösterir. İki nesnenin bağlanması ancak ve ancak eşleşmeyle mümkündür. Yani bir nesnenin çıktısı bir sonraki nesnenin girdisi ile aynı ise bağlanma gerçekleşir aksi durumda sistem bağlanmaya izin vermez. Ayrıca bir nesnenin girdi kısmından başka bir nesneye bağlanma gerçekleşmez. Bağlamanın başlangıç noktası çıktı olmak zorundadır.

3. 6. Gen İfadesi ve Biyokimyasal Reaksiyonlar

3.6.1. Entity Modellemesi

Bir dönüşümü ifade eden entity yapıları içi boş bir dikdörtgen ile modellenir (Şekil 3.36) Bu dikdörtgenin sol tarafında girdiler sağ tarafında çıktı yapıları üstte ise kontrol proteinleri vardır. Entity yapıları biyokimyasal bir dönüşümü modelleyebileceği gibi bir genin ifadesini de modelleyebilir. Bu durumda girdi kısmında ilgili genin ifadesinin başlaması için gerekli proteinler, üstte bu genin ifadesini artıran azaltan kontrol proteinleri ve çıktıda ise ilgili genin ifadesi olan proteinler vardır.



Şekil 3.36 prCell yazılımında entity gösterimi

3. 7. Üyelik ve Temel Veri Kaydı

3.7.1. Üyelik

Yazılım üyelik sistemiyle çalıştığından dolayı öncelikle sisteme üye olmak gerekmektedir. Üyelik formu doldurulduktan sonra sitem mail adresinize şifre gönderecek. Kullanıcı adı ve şifreyle sisteme giriş yapabileceksiniz. Şekil 3.37’de giriş ekranı gösterilmiştir.

The image shows two forms on a light blue background. The top form is titled "Sign In" in orange. It has two input fields: "Email" and "Password", both with placeholder text. Below the fields is a blue button labeled "Enter". The bottom form is titled "Sign Up" in orange. It has two input fields: "UserName" and "Email", both with placeholder text. Below the fields is a blue button labeled "Create".

Şekil 3.37 prCell yazılımında giriş ekranı

3.7.2. Kütüphane Oluşturma

Sisteme girişi yaptıktan sonra sinyal yolağı tasarımlarında kullanacağımız yapıları tanımlamamız gereklidir. Bunun için bir kütüphane oluşturacağız ve kütüphaneye ne kadar fazla entity (Şekil 3.38), molekül (Şekil 3.39), protein (Şekil 3.40), modifiye protein (Şekil 3.41) ve kompleks yapılar(Şekil 3.42) tanımlarsak yazılım o derece yetenekli hale gelecektir.

Kompleks yapılar ve entity'ler oluşturabilmek için öncelikle sisteme protein ve molekül kaydetmemiz şarttır çünkü hem kompleks yapılar hem entity hem de bütün bu nesnelerin girdi ve çıktıları molekül ve proteinlerden oluşmaktadır.

Library

Entity
Molecule
Protein
ProteinModify
Complex

Entity ✕

Symbol

Name

Explanation

Şekil 3.38 prCell yazılımında temel entity kaydı

Yukarıda görüldüğü üzere ilk oluşturulan kayıt entity kaydı. Ancak entity kaydının önceliklenmesi şart değildir. İlk olarak protein ya da molekül kaydı da yapabilirsiniz. Buradaki kayıtlar temel bilgilere göredir. Her entity bir sembolle ifade edilir ve sembol sadece tek bir entity'ye aittir. Entity'le biyokimyasal dönüşümleri veya gen ifadelerini temsil ettiğinden dolayı sembol verirken veya isimlendirirken bu duruma göre isimlendirme ve sembollendirme yapabilirsiniz. Açıklama kısmında ise bu dönüşüm anlatan özet bir bilgi girebilirsiniz.

Library

Entity **Molecule** Protein ProteinModify Complex

Molecule

x

Symbol

Name

Explanation

Şekil 3.39 prCell yazılımında bileşik/molekül temel bilgilerinin kaydı

Sinyal yolağı tasarımının bir diğer temel nesnesi de molekül yapılarıdır. Sinyal yolağında kullanılacak molekül yapıları temel bilgileriyle birlikte bu bölümde sisteme kaydedilir. Entity kaydında olduğu gibi molekülün sembolü ve adı moleküle özeldir ve başka moleküle verilemez.

Library

Entity Molecule Protein ProteinModify Complex

Protein

×

Symbol

Name

Explanation

Save

Şekil 3.40 prCell yazılımında proteinlerin temel bilgi kaydı

Hücredeki işlevsel açıdan en etkili yapı olan proteinler sinyal yolağı tasarımında da en işlevsel nesne olarak karşımıza çıkacaktır. Önceki nesnelerin kaydında olduğu gibi burada da protein sembolü ve protein ismiyle birlikte kaydı yapılır. Genlerin ifadesi olan proteinlerin sembolü ve ismi ilgili genle ilişkili olacak şekilde kaydedilebilir.

Library

Entity
Molecule
Protein
ProteinModify
Complex

ProteinModify ×

Source Protein

Target Protein

Modify select ▼

Explanation

Save

Şekil 3.41 prCell yazılımında modifiye olmuş proteinlerin kaydı

Bir proteinin temel bilgilerini kaydettikten sonra eğer bu protein modifiye olan bir protein ne ile modifiye oluyorsa ise listeden phosphorylation, dephosphorylation, glycosylation, ubiquitination, methylation bir tanesi seçilerek kaydı yapılır. Burumda bir proteinin modifiye hali artık sistem farklı yeni bir protein gibidir ancak sembolü yeni durumda “ () ” alır.

Library

Entity
Molecule
Protein
ProteinModify
Complex

Complex ✕

Symbol

Name

Explanation

Save

Şekil 3.42 prCell yazılımında protein komplekslerinin temel bilgilerinin kaydedilmesi

Çoğu zaman proteinler tek başlarına işlev göremezler. Diğer protein ve moleküllerle birleşerek kompleks yapılar oluştururlar. Bu kompleks yapılar işlevsel hale geldikten sonra sinyal yolları içinde aktif rol oynayabilirler. Yukarıda gördüğümüz ekranda bu kompleks yapıların sadece temel bilgileri kaydedilir. Bu aşamadan sonra da kompleks yapının barındıracağı proteinler ve moleküller belirlenir.

3.7.3. Kompleks Yapı Oluşturma

Kütüphane oluşturduktan sonra tasarımda kullanacağımız protein komplekslerini de tanımlayabiliriz. Kütüphane kısmına kompleksin adı ve sembolü tanımladıktan sonra hangi yapıları barındıracaksa onları ekleriz ve sisteme kaydederiz.

Aşağıdaki ekran görüntüsünde kırmızı renkle gösterilen alanda daha önce temel bilgileriyle kaydetmiş olduğumuz kompleks yapının sembolü girilerek ekleme butonuna basılır. Sistemde kayıtlı sembole ait kompleks yapıyı hemen altında sembol ve ismiyle birlikte

gördüğümüz listeye ekler. Buraya sadece bir kayıt ekleyebiliriz. İkinci kompleks yapıyı buraya eklememize sistem izin vermeyecektir. Çünkü burada sadece bir tane kompleks yapının bileşenlerini oluşturabiliriz. Aynı anda iki kompleks yapıyı oluşturamayız. Eğer eklemek istediğimiz kompleks yapı sistemde kayıtlı değilse bulamayacaktır ve ekleyemeyecektir. Bu durumda kütüphane kısmına giderek ilgili kompleks yapının adını ve sembolünü kaydetmemiz gerekecek. Bu aşamayı bitirdikten sonra bu kompleks yapının hangi bileşenlerden oluştuğunu biliyorsak ilgili bileşenleri sırasıyla eklememiz gerekecektir. Şekil 3.43’de kaydın nasıl yapıldığı gösterilmiştir.

Complex Item

Symbol

Add

Symbol Name

Protein

Symbol

Add

Symbol Name

Compound

Symbol

Add

Symbol Name

Şekil 3.43 prCell yazılımında protein komplekslerinin bileşenlerinin oluşturulması

Kompleks yapılar içinde proteinleri ve bazı molekülleri barındırabilir. Burada hangi protein veya proteinleri barındırıyor ise teker teker ekleriz. Solda gördüğümüz protein kısmında proteinin sembolünü yazarak sistemde böyle bir protein varsa hemen altındaki sembol ve isim olan listeye eklemesini sağlarız. Eğer böyle bir protein eklenemezse sistemde kaydı yok demektir. Library kısmına giderek burada ilgili proteinin girişini yapmak gerekir. Bu aşamada birden fazla protein ekleyebiliriz çünkü kompleks yapılar birden fazla protein bulundurabilir. Yanlış olduğunu düşündüğümüz bir protein eklersek de eklediğimiz kaydın hemen sonunda “delete” yazısına tıklayarak listeden çıkartabiliriz.

Üstteki resimde sağ tarafta gördüğümüz compound alanında ise kompleks yapı aynı zamanda bileşik veya bileşikler de içeriyorsa bu kısımdan ekleriz. Bileşiğin sembolü yazılarak butona basıldığında sistem kayıtlı olan böyle bir bileşiğin var olup olmadığını tarayacak ve varsa

hemen altındaki listeye ekleyecektir. Burada da yine birden fazla bileşiği ekleyebiliriz. Eklediğimiz bileşiği istersek çıkarabiliriz.

Kompleks yapının artık hangi bileşenlerden oluştuğuna karar verdiyse, sisteme kaydetmek için sağ alt taraftaki “save” butonuna basarız. Bundan sonra diagram ortamına bu kompleksin sembolüyle çağrıldığında içinde eklediğimiz protein ve moleküllerle birlikte gelecektir.

3.8. Port Ekleme

3.8.1. Nesnelere Giriş ve Kontrol Portlarının Eklenmesi

Nesnelerin birbirleriyle olan etkileşimlerini gösterirken portları kullanırız. Bir nesnenin giriş kısmında bir portun olması aslında onu etkileyen bir başka nesnenin var olduğu anlamını taşır. Portların üzerindeki sembol isminden o portun hangi tasarım nesnesini temsil ettiğini anlayabiliriz. Port bazen bir protein bazen bir bileşik bazen ise kompleks bir yapıyı temsil edebilir. Entity yapılarının üst kısmında görünen portlar ise o dönüşümü kontrol eden protein ve kompleks yapıları simgeler.[43]

Portların aldığı renklere göre anlamları vardır. Örneğin bir portun rengi yeşilse bu port nesnesi bağlı olduğu nesneyi aktive ettiği manasına gelir. Eğer bir portun rengi kırmızı ise bu da port elemanın bağlı olduğu yapıyı inhibe ettiği manasına gelir. Aşağıda hangi rengin ne anlama geldiği yazılmıştır.

Activation	:	yeşil
Inhibition	:	kırmızı
Expression	:	mavi
Repression	:	turuncu
IndirectEffect	:	gri
Statechange	:	mor
Association	:	sarı
Dissociation	:	siyah

Moleküllere Port Ekleme

Molekül(compound) yapıları aktive eden ya da inhibe eden diğer molekül veya proteinlerin molekül üzerinde portlarla gösteririz. Bir moleküle sistem üzerinden port eklemek aslında o molekülü hangi yapıların nasıl etkilediğini belirtmek demektir. Temel bilgilerle kaydedilen molekül, gerekli portlar eklendikten sonra son halini alır. Şekil 3.44’de bileşiğe nasıl port eklendiği gösterilmiştir.

Compound Item

Symbol

Effect act inh exp repres ind.eff st.cha

Symbol Name Explanation

Protein Port

Symbol

Effect act inh exp repres ind.eff st.cha

Symbol Name Activeness

Compound Port

Effect act inh exp repres ind.eff st.cha

Symbol Name Activeness

Complex Port

Symbol

Effect act inh exp repres ind.eff st.cha

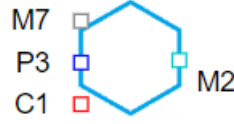
Symbol Name Activeness

Şekil 3.44 prCell yazılımında bileşik/molekül port ekleme



Şekil 3.45 prCell yazılımında bileşik/molekül port eklenmemiş hali.

Hiç port eklenmezse bütün nesnelere olduğu gibi bileşikler/moleküller de sadece çıkış portunu(Bknz Şekil 3.45) otomatik olarak alır ve bu portun simgesi diğer bütün nesnelere olduğu gibi bileşiğin/molekülün simgesiyle aynıdır .Giriş portları eklendiği durumda ise bu proteini hangi yapılar etkiliyorsa,etki şekline göre port renkleri belirlenmek suretiyle ilgili portlar Şekil 3.46’da görüldüğü gibi eklenir.



Şekil 3.46 prCell yazılımında bileşik/molekül giriş portu olarak M7,P3,C1 almış. Çıkış portundan da anlaşılacağı üzere bu bileşiğin/molekülün simgesi M2dir.

Proteinlere Port Ekleme

Metabolizmada bir protein bazen işlev görebilmesi için başka proteinlerle aktive edilmesi gerekebilir. Bununla birlikte bu proteini inhibe eden yapılar da olabilir. Bir proteine ait olumlu veya olumsuz etki eden yapıları ilgili proteine port ekleyerek (Şekil 3.47) göstermemiz gerekir. Bütün portları eklenmiş ve bu şekilde sisteme kaydedilmiş protein artık sinyal yolağı tasarımında kullanılabilir demektir.

Protein Item

Symbol

Symbol	Name	Explanation
--------	------	-------------

Protein Port

Symbol

Effect act inh exp repres ind.eff st.cha

Symbol	Name	Activeness
--------	------	------------

Compound Port

Symbol

Effect act inh exp repres ind.eff st.cha

Symbol	Name	Activeness
--------	------	------------

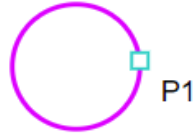
Complex Port

Symbol

Effect act inh exp repres ind.eff st.cha

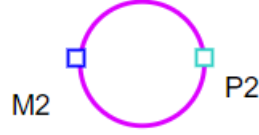
Symbol	Name	Activeness
--------	------	------------

Şekil 3.47 prCell yazılımında proteinlere port ekleme



Şekil 3.48 prCell yazılımında port eklenmemiş protein

Şekil 3.48'den anlaşıldığı üzere P1 proteini etkileyen herhangi bir nesne yoktur.



Şekil 3.49 prCell yazılımında giriş portu olarak M2 molekülü eklenmiş P2 proteini.

Şekil 3.49'dan görüldüğü üzere mavi renk olması M2 molekülünün P2 proteinine etkisinin "Expression" olduğu anlamına gelir

Kompleks Yapılara Port Ekleme

Proteinler çoğu zaman tek başına işlev görmezler. Diğer molekül ve proteinlerle birlikte kompleks yapılar oluştururlar. Bu yapıları da yine başka proteinler veya moleküller aktive ya da inhibe edebilirler. Bir kompleks yapıya etki eden protein veya molekülleri ilgili kompleks yapıya ekleyeceğimiz portlarla gösteririz.(Şekil 3.50)

Complex Item

Symbol

Add

Symbol Name

Protein Port

Symbol

Effect act inh exp repres ind.eff st.cha

Add

Symbol Name Activeness

Compound Port

Symbol

Effect act inh exp repres ind.eff st.cha

Add

Symbol Name Activeness

Complex Port

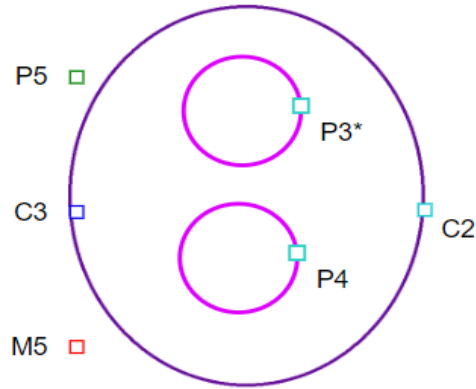
Symbol

Effect act inh exp repres ind.eff st.cha

Add

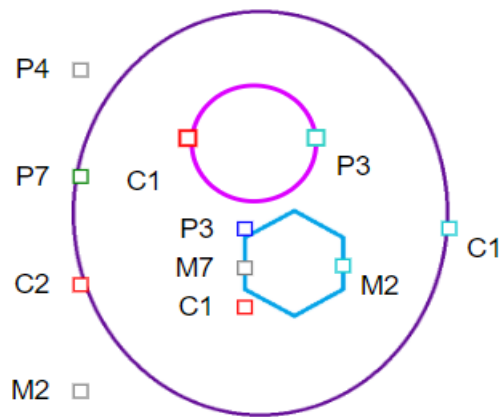
Symbol Name Activeness

Şekil 3.50 prCell yazılımında protein komplekslerine port eklenmesi



Şekil 3.51 prCell yazılımında port eklenmemiş protein kompleksi.

Şekil 3.51'e baktığımızda P3*(P3 proteinin modifiye olmuş hali) ve P4 proteinin bileşenlerinden oluşan bu kompleks yapının giriş portlarındaki P5 proteini (yeşil renkte aktive olduğu için kompleksi aktive eder) C3 kompleksi mavi renkte ve M5 molekülü kırmızı renkte (C2 kompleksini inhibe ettiği anlaşılıyor) olduğu anlamını çıkartabiliriz. Şekil 3.52'de ise farklı bir kompleks yapısı gösterilmiştir.



Şekil 3.52 prCell yazılımında port eklenmiş C1 kompleksi P3 proteini ve M2 bileşiği/molekülünden meydana gelmektedir.

Entity Yapısına Port Ekleme

Entity yapısı daha önce belirttiğimiz üzere bir dönüşümü ifade eder. Bu dönüşüm bir biyokimyasal reaksiyon olduğu gibi bir genin ifadesi de olabilir. Dönüşümün girdileri hangi molekül veya protein ise bunlar entity port olarak eklenir. Entity'i kontrol eden proteinler de üst kısma port olarak eklenir. Son olarak da bu dönüşüm sonucu oluşan yapılar port olarak eklenir. Bütün bunlar yazılım içinde ilgili sayfada yapılır ve sisteme eklenir (Şekil 3.53).

Entity

Symbol

Symbol **Name** **Explanation**

Control Protein

Symbol

Effect activation inhibition indirect effect

Name **Symbol** **Explanation**

Control Complex

Symbol

Effect activation inhibition indirect effect

Symbol **Name** **Explanation**

Input Compound

Symbol

Symbol **Name** **Explanation**

Output Compound

Symbol

Symbol **Name** **Explanation**

Input Protein

Symbol

Symbol **Name** **Explanation**

Output Protein

Symbol

Symbol **Name** **Explanation**

Input Complex

Symbol

Symbol **Name** **Explanation**

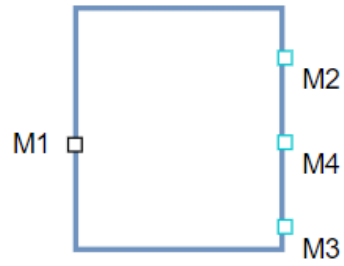
OutputComplex

Symbol

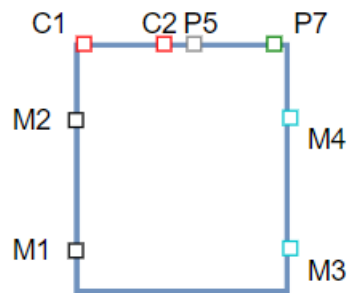
Symbol **Name** **Explanation**

Şekil 3.53 prCell yazılımında entity yapısına port eklemek.

Entity'nin input kısmı bileşik/molekül, protein ve protein kompleksi olabileceği gibi çıkış portları da bileşik/molekül, protein ve protein kompleksi olabilir. Kontrol portları ise protein veya protein kompleksi olabilir sadece. Şekil 3.54 ve Şekil 3.55 port eklenmiş entity yapısı modellenmiştir.



Şekil 3.54 prCell yazılımında entity'ye port eklenmesi.



Şekil 3.55 Kontrol portları olan bir entity yapısı.

Burada entity'nin girdi kısmında M1ve M2 bileşiği/molekülü varken bu dönüşümü kontrol eden yapılar C1ve C2 kompleksi ile P5 ve P7 proteindir. Bu dönüşüm sonucunda M3 ve M4 molekülleri oluşmuştur.C1 ve C2 kompleks yapıları bu dönüşümü olumsuz etkilerken P7 olumlu yönde etkilemektedir.P5 ise dolaylı etki gösterdiği anlaşılmaktadır.

4.BULGULAR

Şimdiye kadar prCell'in hangi temellere dayandığını, veri kayıt ekranlarını, bileşik, protein, protein kompleksleri ve entity yapılarına nasıl port eklendiğini gördük. Bu bölümde ise adım adım prCell'in nasıl kullanılacağını göreceğiz ve yapay bir sinyal yolağı ile gerçek bir sinyal yolağının prCell üzerinde oluşturulmuş halini göreceğiz.

4.1. Tasarım

Sinyal yolağı tasarlayabilmek için kütüphanemizde bileşik, protein, protein kompleksleri ve entity yapılarının olduğunu varsayarak, öncelikle proje kaydının yapılması, arkasından diagram ekranındaki sol menüden sisteme kayıtlı nesnelere bularak tasarım ekranına atılmasını sağlıyoruz. Bu aşamadan sonra giriş çıkış eşleşme kuralına uyanları birbirine bağlıyoruz.

4.1.1. Proje Oluşturma

Sinyal yolağı tasarımına başlamadan önce bir proje kaydı oluşturmamız gerekir. (Şekil 4.1) Proje kaydında projenin adı ve sembolünü sisteme kaydettikten sonra bu proje üzerinden tasarım ortamına geçeriz. Tasarım ortamında ise bir sonraki adım olan diagramı çizmeye başlarız.

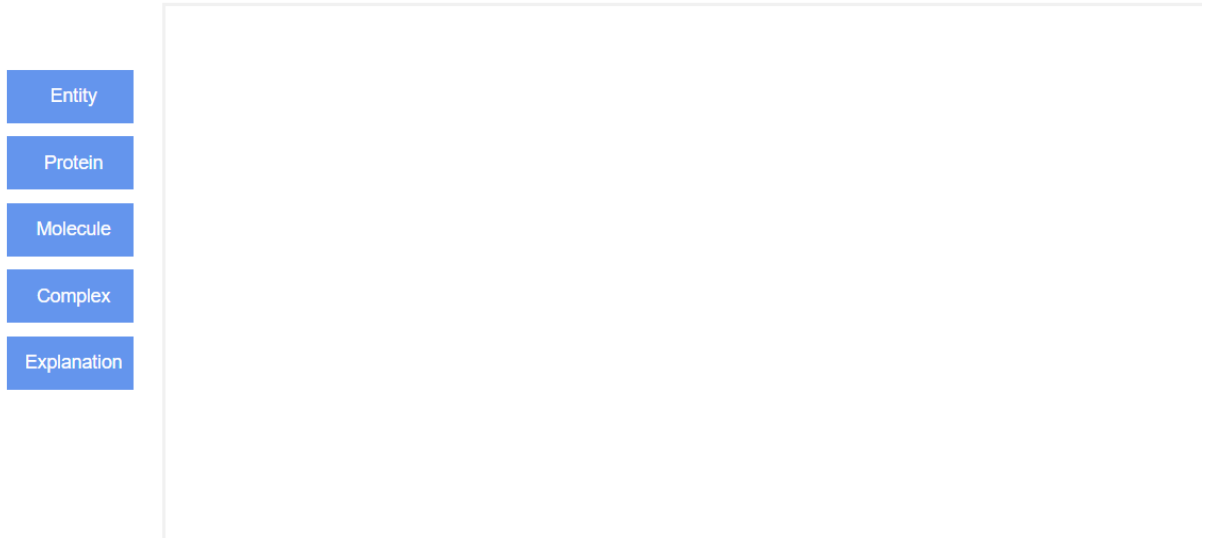
New Project

ProSymbol	<input type="text"/>
ProjectName	<input type="text"/>
Explanation	<input type="text"/>

Şekil 4.1 prCell yazılımında temel bilgilerle proje kaydı.

4.1.2. Diagram Çizme

Molekülleri, proteinleri kompleks yapıları ve entity yapılarını girdi ve çıktılarıyla birlikte tanımladıktan sonra artık sinyal yolağı tasarlayabiliriz. Bunun için oluşturduğumuz projenin tasarım sayfasına gideriz. Sol kenardaki menüden diagram ortamına ilgili molekül protein ve entity'leri atarak girdi-çıkıı eşleşmesi mantığına göre bağlarız. Bu şekilde soldan sağa doğru sinyal yolağının akışı sağlanır. Şekil 4.2'de tasarım ekranını gösterilmiştir.

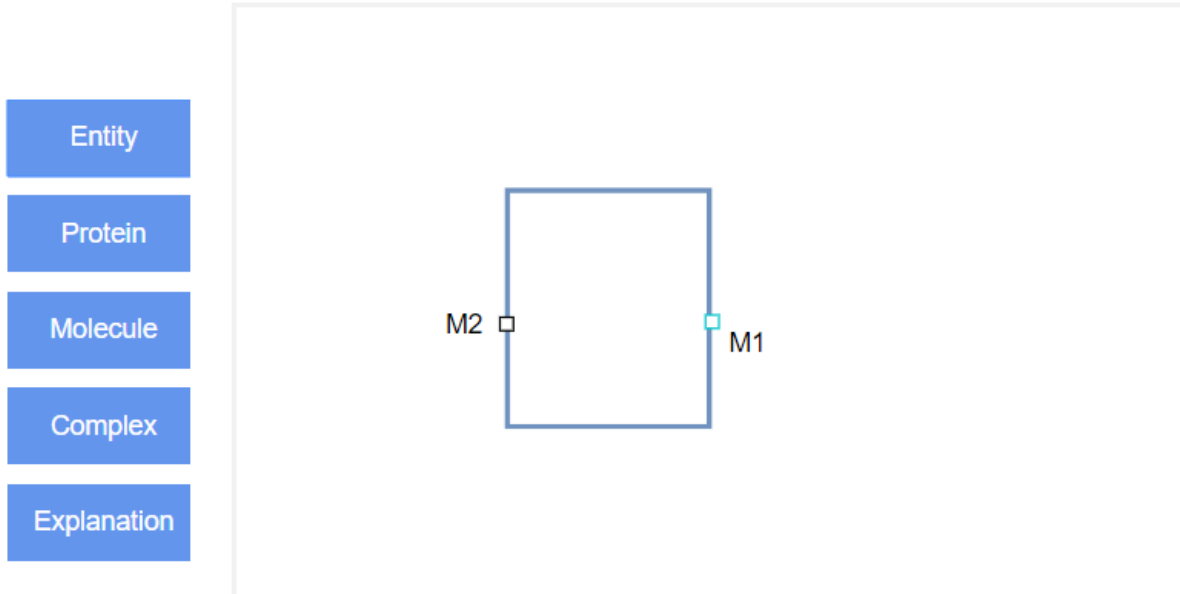


Şekil 4.2 prCell yazılımında tasarım ekranı.

Bu ekran scrolla genişletilebildiğinden bilgisayar ekranının yaklaşık 300 katı kadardır ve çok büyük sinyal yolağı tasarımları dahi yapılabilir. Soldaki menüden tasarım ortamında kayıtlı verilerinizden nesnelere oluşturabilirsiniz.

Entity ekleme

Şekil 4.3 prCell yazılımında tasarım ortamına entity atmak.



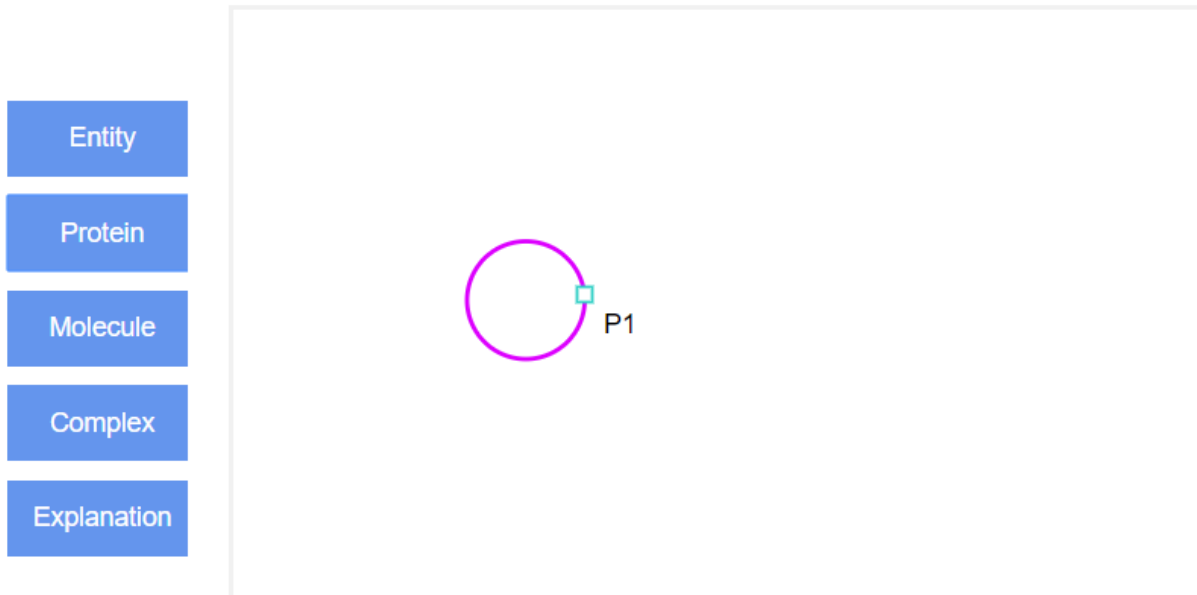
Şekil 4.4 prCell yazılımında tasarım alanına eklenmiş entity

Entity'lerde diğer nesnelere olduğu gibi çıkış portu sadece bir tane değildir. Ayrıca diğer nesnelere aksine çıkış portu hiç bir zaman entity'nin adını almaz. Çıkış portları yeni oluşan, elde edilen proteini veya bileşiklerini gösterir. Entity bir değişimi, dönüşümü simgeler. Aslında elektronik devrelerdeki mantık kapılarına da benzetebiliriz. Ancak entity mantık kapıları gibi sabit 14 çeşit değildir. Giriş, çıkış ve kontrol portlarını kullanıcı belirlediği için sınırsızdır. Bu durum sinyal yolağı programlamada kullanıcıya büyük bir esneklik kazandıracaktır.

Protein ekleme

Şekil 4.5 prCell yazılımında tasarım ortamına protein atmak.

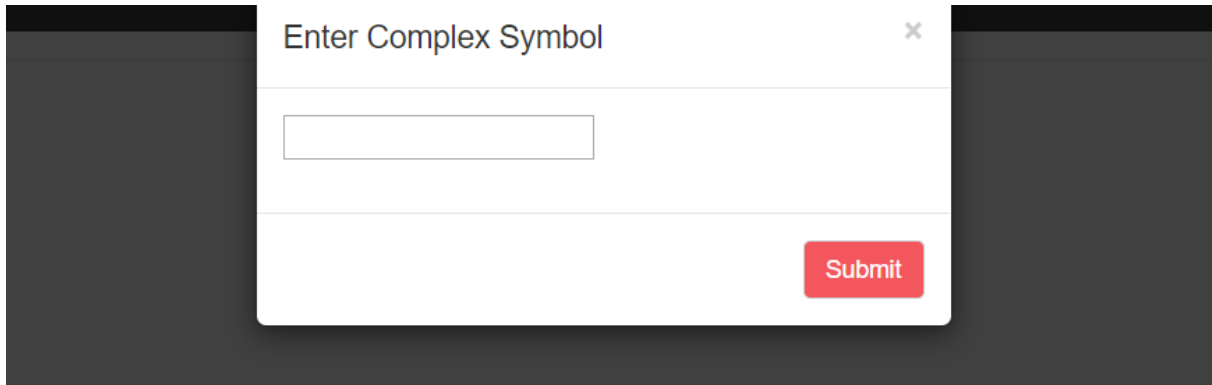
Ekranda görülen kısma proteinin sembolünü yazarak butona tıkladığınızda eğer protein sistemde kayıtlı ise tasarım ekranında varsa portlarıyla birlikte oluşacaktır.



Şekil 4.6 prCell yazılımında giriş portu olmayan P1 proteini eklenmiş.

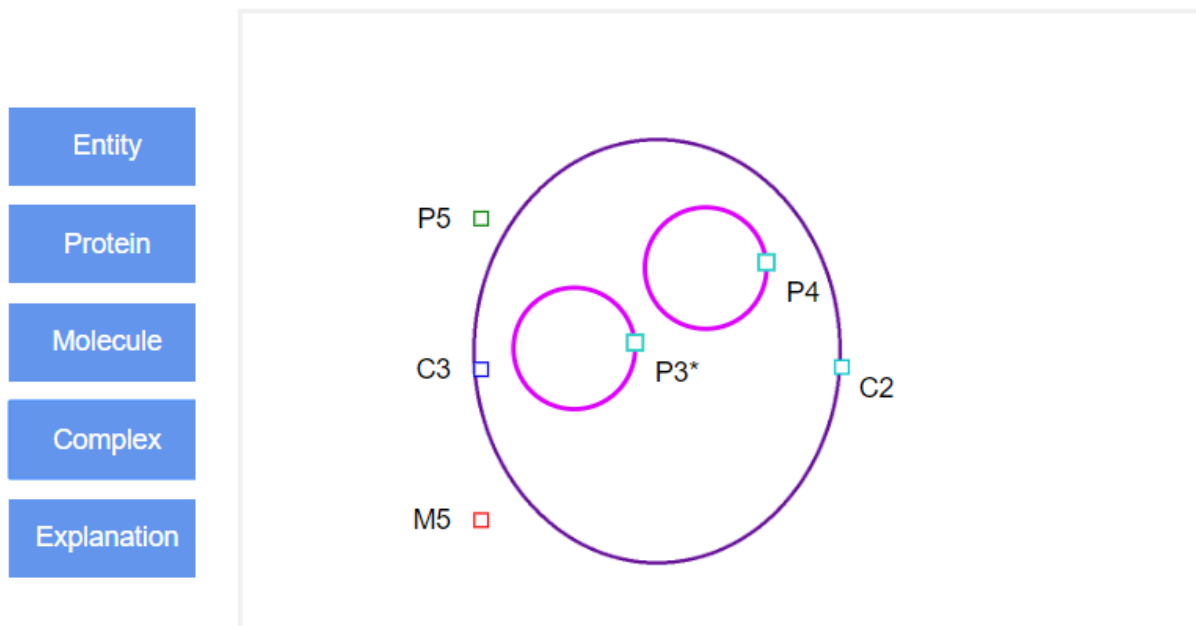
Şekil 4.5 ve Şekil 4.6'da tasarım ortamında bir proteinin nasıl oluşturulacağını görüyoruz

Kompleks yapı ekleme



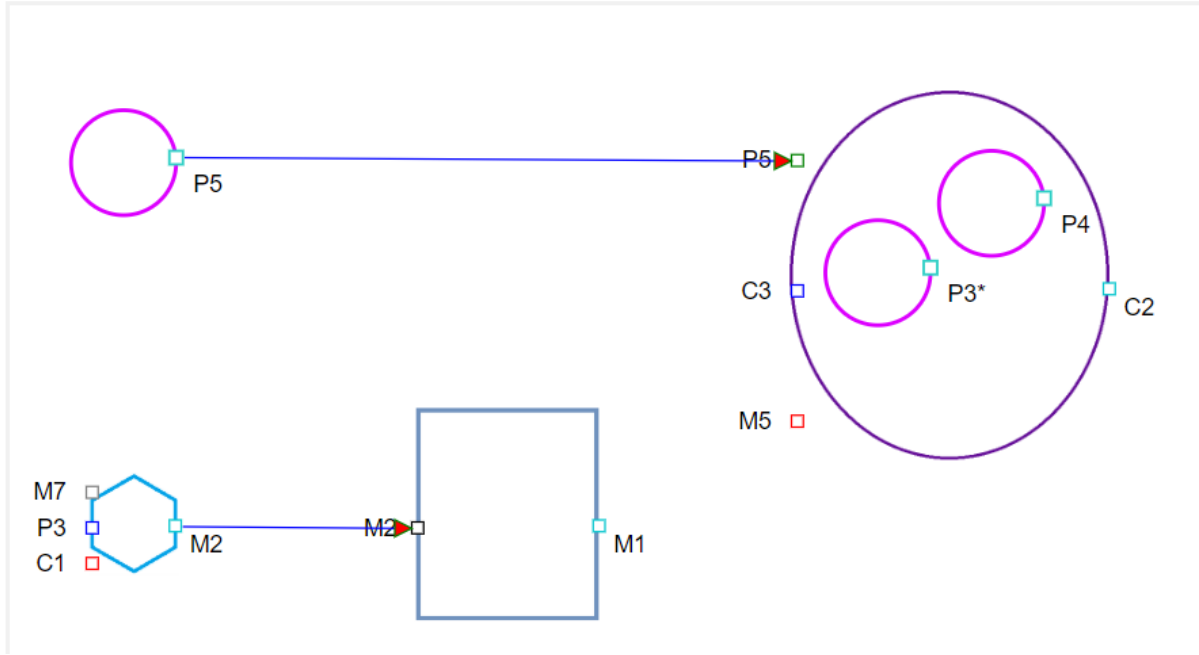
Şekil 4.7 prCell yazılımında tasarım ortamına protein kompleksi atmak.

Yine aynı şekilde tasarım ortamına protein kompleksi eklemek istersek protein kompleksinin sembolünü yazarak butona tıkladığımızda, sistemde kayıtlıysa hem portlarıyla hemde içinde barındırdığı protein veya bileşiklerle birlikte tasarım ortamında oluşacaktır. (Bknz. Şekil 4.7 ve Şekil 4.8)



Şekil 4.8 prCell yazılımında tasarım ortamında oluşturulmuş C2 kompleks yapısı.

Nesnelerin Girdi Çıktı Eşleşme Kuralına Göre Bağlanması



Şekil 4.9 prCell yazılımında tasarım ortamına bağlanma kuralının uygulanması.

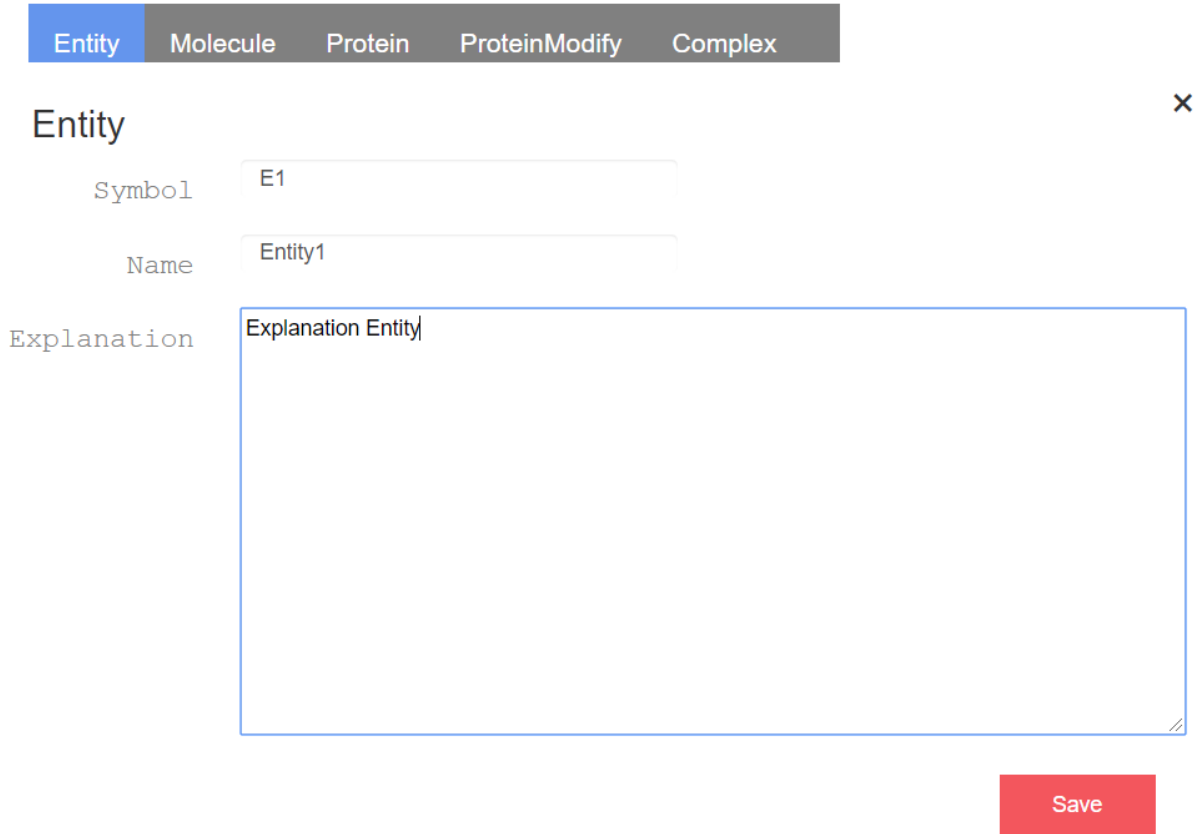
P5 proteinin çıkış portu kendisi aynı zamanda C2 protein kompleksini aktive ediyor(port rengi yeşil).Portlar isimleri tam olarak aynı olduğu için burada bağlanma oluşur.(Bknz. Şekil 4.9)Ancak P5 proteini C2 kompleksinin diğer portları olan C3 ve M5 portlarına bağlamaya çalışsak dahi bağlanma gerçekleşmeyecektir.Bağlanma hiç bir zaman bir nesnenin giriş portundan başlamaz.M2 bileşiğinin giriş portlarından birinden bağlanma gerçekleştirmek için çizgi çizmeye kalsak bu çizgi hiç oluşmayacaktır.[44]

4.2. Örnek Bir Uygulama

4.2.1. Kütüphaneye Veri Kaydı

Uygulamaya üye olduktan sonra yapmamız gereken öncelikle kütüphane oluşturmaktır.Kütüphane bizim temel verilerimizi kaydedildiği yerdir.Protein, modifiye protein, bileşik, entity ve kompleks yapıları temel bilgileriyle bu kısma kaydederiz.Kütüphanemiz veri açısından ne kadar zengin olursa oluşturacağımız etkileşimler ve dolayısıyla sinyal yolağı o kadar çeşitli olacaktır.Şekil 4.10'da entity kaydının sembol, isim ve açıklama gibi temel bilgilerle nasıl yapıldığını görüyorsunuz. Şekil 4.11'de temel bilgileriyle

bileşiklerin kaydedilmesini, Şekil 4.12’de ve Şekil 4.14’de sırasıyla protein ve kompleks yapıların kayıt işlemleri yapılmaktadır. Şekil 4.13’de ise modifiye protein kaydı yapılır ancak modifiye protein kaydı yapılırken modifiye olan ve modifiye eden proteinlerin daha önce kütüphanede kayıtlı olması gerekir. Ayrıca “Modify” alanında çıkan listeden modifikasyonun türü seçilmelidir.



The screenshot displays the prCell software interface for creating a new Entity. At the top, there is a navigation bar with tabs for 'Entity', 'Molecule', 'Protein', 'ProteinModify', and 'Complex'. The 'Entity' tab is currently selected. Below the navigation bar, the title 'Entity' is displayed on the left, and a close button (X) is on the right. The form consists of three main input fields: 'Symbol' with the value 'E1', 'Name' with the value 'Entity1', and 'Explanation' with the value 'Explanation Entity'. A large red 'Save' button is located at the bottom right of the form.

Şekil 4.10 prCell yazılımında E1 entity’sinin kaydedilmesi

Library

Entity	Molecule	Protein	ProteinModify	Complex
--------	----------	---------	---------------	---------

Molecule

Symbol

Name

Explanation

Şekil 4.11 prCell yazılımında M1 molekülünün/bileşiğinin kaydedilmesi

Library

Entity	Molecule	Protein	ProteinModify	Complex
--------	----------	---------	---------------	---------

Protein

Symbol

Name

Explanation

Şekil 4.12 prCell yazılımında P1 proteininin kaydedilmesi.

Library

Entity Molecule Protein ProteinModify Complex

ProteinModify ×

Source Protein

Target Protein

Modify

Explanation

Şekil 4.13 prCell yazılımında P2 proteininin P3 tarafından modifiye edilmesinin kaydedilmesi.

Library

Entity Molecule Protein ProteinModify Complex

Complex ×

Symbol

Name

Explanation

Şekil 4.14 prCell yazılımında C1 kompleks yapısının kaydedilmesi.

Complex Item

Symbol

Add

ComplexSymbol ComplexName

c3 Complex3 del

Protein

Symbol

Add

ProteinSymbol ProteinName

P4 Protein4 del

Compound

Symbol

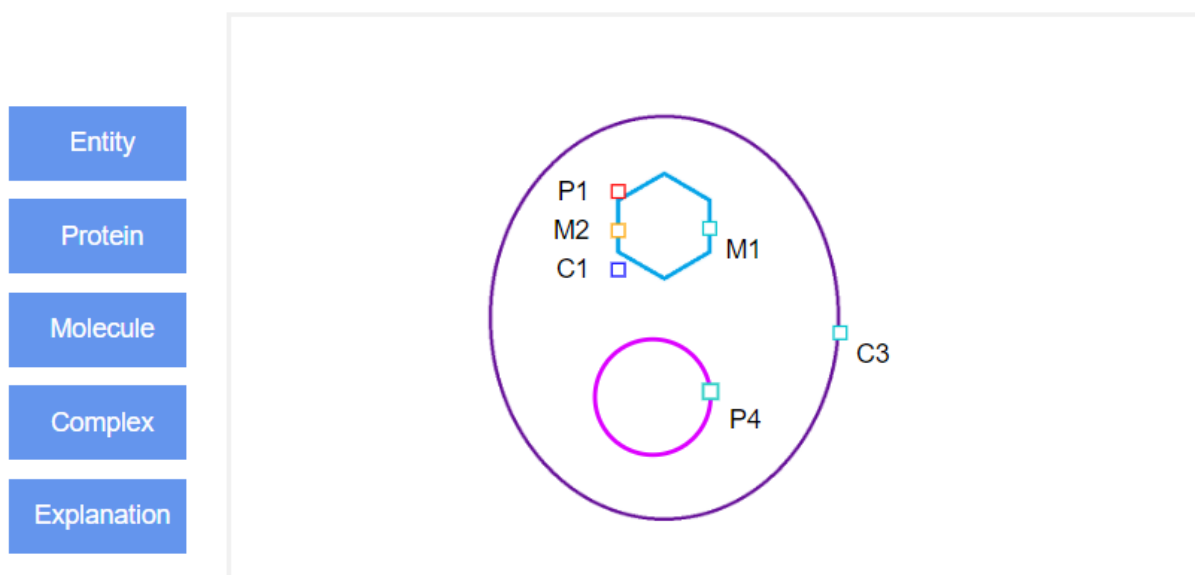
Add

ComponentSymbol ComponentName

M1 molecule1 del

Şekil 4.15 prCell yazılımında C3 protein kompleksinin bileşenlerinin oluşturulması.

Yukarıda görülen kısımda kırmızı bölümde hangi kompleksin içeriğini belirleyeceksek o kompleksi ekliyoruz. Bu kısma istesek de ikinci bir kayıt ekleyemeyiz. Çünkü aynı anda sadece bir kompleksin bileşenlerini belirleyebiliriz. Bu kompleksin bileşenlerinde hangi protein veya proteinler varsa sol alt kısımdaki protein ekleme kısmını kullanırız. Birden fazla protein eklenebilir. (Bknz. Şekil 4.15) Sağ alt kısımda ise hangi bileşikler varsa yapısında onları ekleriz. Burada da birden fazla eklem yapabiliriz. Ancak bütün ekleyeceğimiz yapıların daha önce sistemde kaydının bulunması gerekir. Bu nesnelere portlarıyla birlikte tasarım kompleksinin içinde ortamında oluşacaktır. (Bknz. Şekil 4.16)



Şekil 4.16 prCell yazılımında oluşturulan C3 kompleksinin gösterilmesi

4.2.2. Nesnelere Giriş Ve Kontrol Portlarının Eklenmesi

Compound Item

Symbol

Effect act inh exp repres ind.eff st.cha

ItemSymbol	ItemName	Explanation	del
M1	molecule1	dcqfq	del

Protein Port

Symbol

Effect act inh exp repres ind.eff st.cha

ProteinSymbol	ProteinName	Active	del
P1	Protein1	3	del

Compound Port

Effect act inh exp repres ind.eff st.cha

ComponentSymbol	ComponentName	Active	del
M2	molecule2	1	del

Complex Port

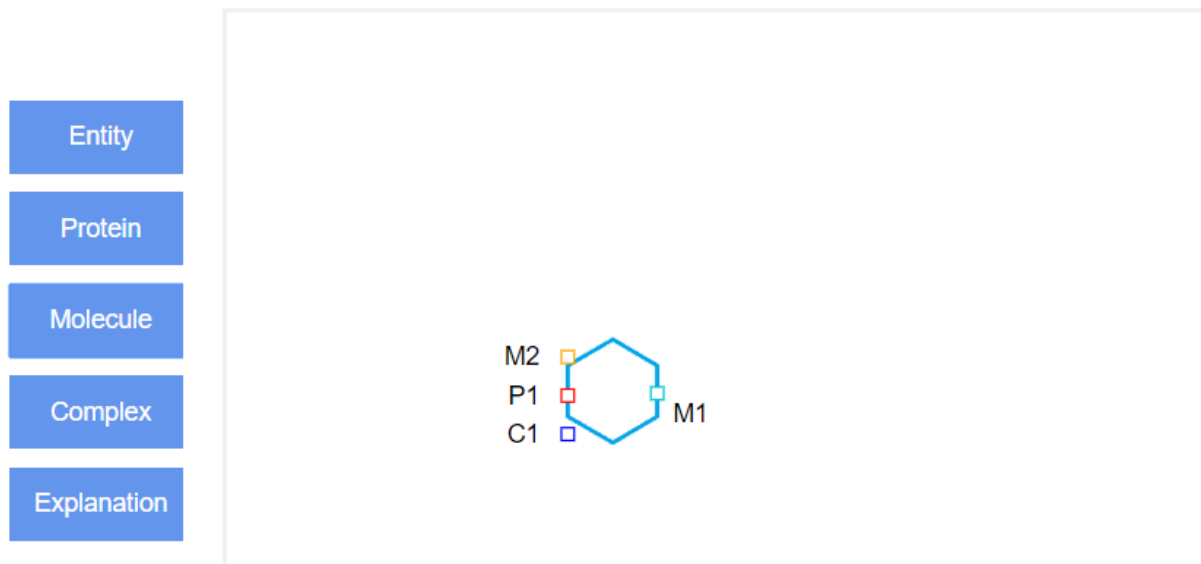
Symbol

Effect act inh exp repres ind.eff st.cha

ComplexSymbol	ComplexName	Active	del
C1	Complex	4	del

Şekil 4.17 prCell yazılımında M1 molekülüne/bileşiğine P1,M2 ve C1 portlarının eklenmesi

Buradaki amaç ise M1 bileşiğine etkileşimde olduğu yapıları etki türüyle birlikte belirlemektir.Etki türleri portun rengini belirlemeye yarayacaktır.(Bknz. Şekil 4.17 ve Şekil 4.18)



Şekil 4.18 prCell yazılımında oluşturulan M1 molekülünün/bileşiğinin gösterilmesi

Protein Item

Symbol

ItemSymbol	ItemName	Explanation	
P2	Protein2	Explanation2	delete

Protein Port

Symbol

Effect act inh exp repres ind.eff st.cha

ProteinSymbol	ProteinName	Active	
P1	Protein1	3	del

Compound Port

Symbol

Effect act inh exp repres ind.eff st.cha

ComponentSymbol	ComponentName	Active	
M3	molecule3	2	delete
M4	molecule4	5	delete

Complex Port

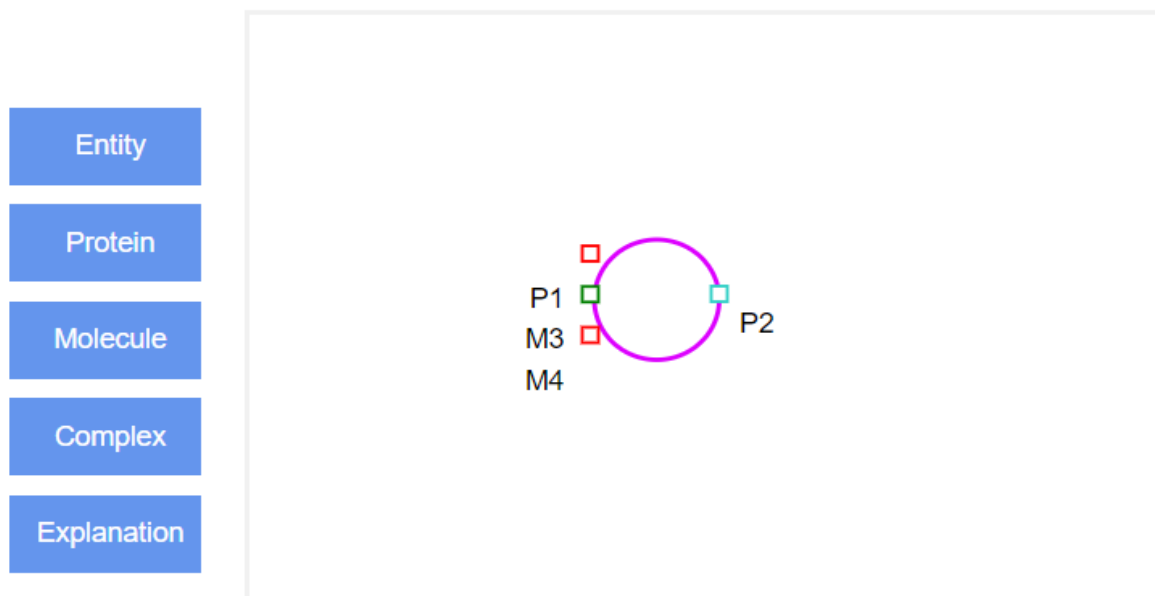
Symbol

Effect act inh exp repres ind.eff st.cha

ComplexSymbol	ComplexName	Active

Şekil 4.19 prCell yazılımında P2 proteinine M3,M4 ve P1 portlarının eklenmesi

Aynı şekilde P2 proteininin etkileşimde olduğu nesnelere kaydı yapılıyor. Bu kayıtlar tasarım ortamında P2 proteini oluştururken portları ve portların renklerini belirleyecektir. Şekil 4.19 ve Şekil 4.20’de bu aşamaların nasıl yapıldığı gösterilmiştir.



Şekil 4.20 prCell yazılımında eklenen P2 proteininin gösterilmesi

Complex Item

Symbol

ItemSymbol	ItemName	del
C1	Complex	

Protein Port

Symbol

Effect act inh exp repres ind.eff st.cha

Symbol	Name	Active	del
P4	Protein4	4	

Molecule Port

Symbol

Effect act inh exp repres ind.eff st.cha

Symbol	Name	Active	del
M2	molecule2	5	

Complex Port

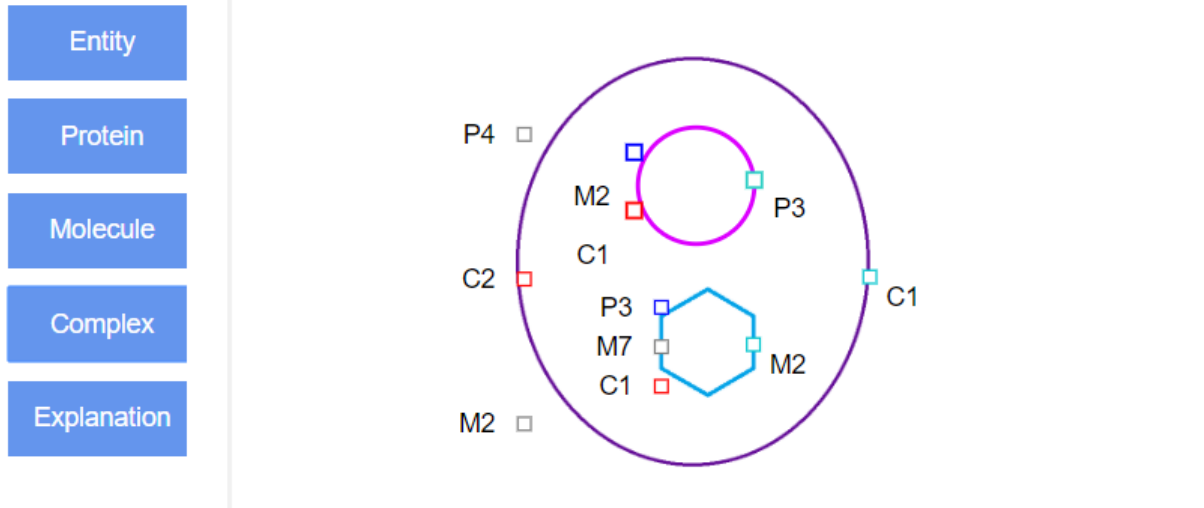
Symbol

Effect act inh exp repres ind.eff st.cha

Symbol	Name	Active	del
C2	Complex2	2	

Şekil 4.21 prCell yazılımında C1 kompleksine P4,M2 ve C2 portlarının eklenmesi

Bu kısımda C1 protein kompleksinin bileşenleri değil giriş portları belirlenmektedir.Kompleksin yapının etkilendiği nesnelerin eklenmesi ve sonucunda oluşan kompleks yapının yeni halini Şekil 4.21 ve Şekil 4.22’de gösterilmiştir.



Şekil 4.22 prCell yazılımında C1 kompleksinin gösterimi

EntitySymbol **Add****Symbol** **Name** **Explanation**

E1 Entity1 Explanation1 delete

Control ProteinSymbol Effect activation inhibition indirect effect**Add****Name** **Symbol** **Explanation**

Protein3 P3 Explanation3 delete

Control ComplexSymbol Effect activation inhibition indirect effect**Add****Symbol** **Name** **Explanation****Input Compound**Symbol **Add****Symbol** **Name** **Explanation****Output Compound**Symbol **Add****Symbol** **Name** **Explanation****Input Protein**Symbol **Add****Symbol** **Name** **Explanation**

P4 Protein4 Explanation4 del

Output ProteinSymbol **Add****Symbol** **Name** **Explanation**

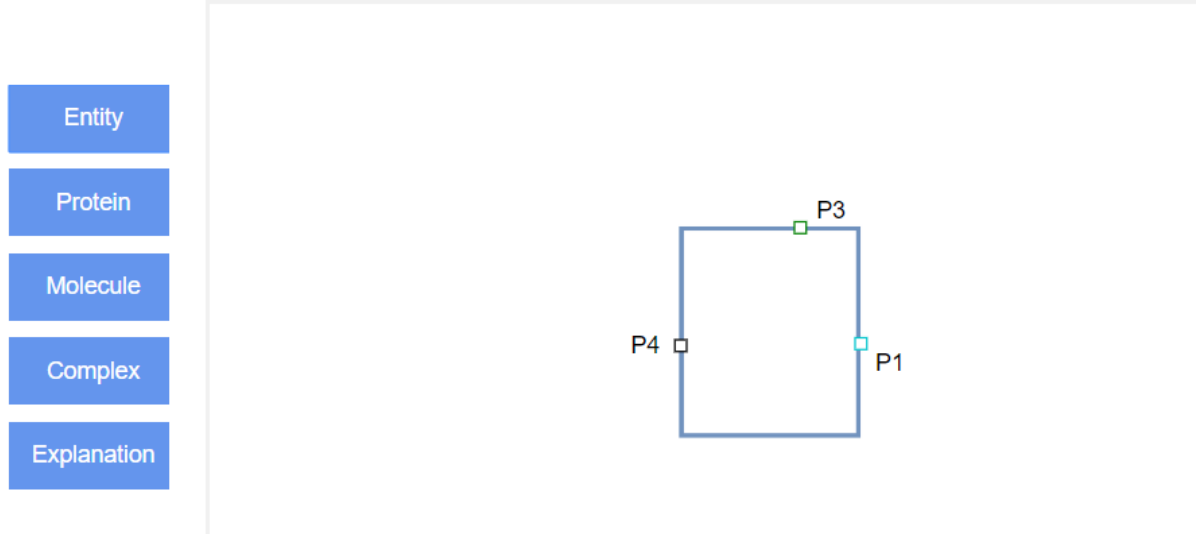
P1 Protein1 Explanation1 delete

Input ComplexSymbol **Add****Symbol** **Name** **Explanation****OutputComplex**Symbol **Add****Symbol** **Name** **Explanation****Save****Şekil 4.23** prCell yazılımında E1 entity'si

Şekil 4.23'de prCell yazılımında E1 entity'sine kontrol proteini olarak P3 girdi proteini olarak P4 çıktı proteini olarak da P1 portlarının eklenmesi gösterilmiştir.

Bu kısımda bir nevi değişimi ve dönüşümü sağlayan entity nesnesinin, bu değişim ve dönüşümün girdilerinin ne olduğunu, değişimi olumlu yönde etkileyen veya olumsuz yönde

etkileyen kontrol proteinlerini ve bu deęişim ve dönüşüm sonunda hangi protein,bileşik veya protein komplekslerini elde ettiğimizi belirleyerek bir model olarak sisteme tanıtırız. Şekil 4.23 ve Şekil 4.24 entitynin oluşturulma aşaması ve kullanma aşaması gösterilmiştir.



Şekil 4.24 prCell yazılımında E1 entity'sinin gösterimi

Daha önce sisteme tanıttığımız E1 entity'sinin tasarım ekranındaki gösterimini görüyoruz

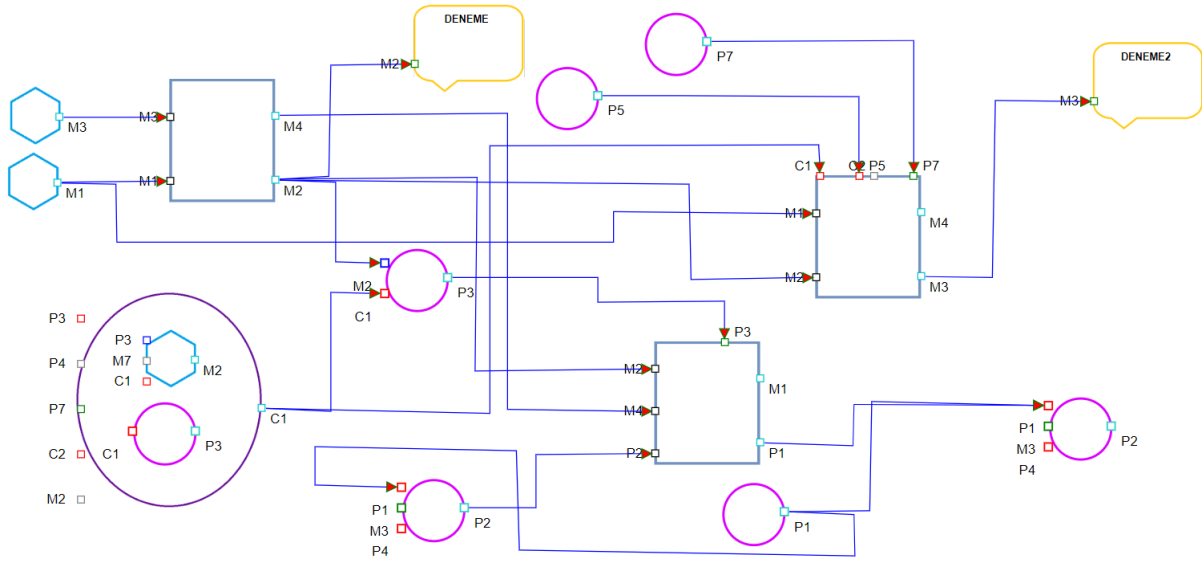
4.2.3. Proje Oluşturma ve Diagram Çizme

New Project

ProSymbol	<input type="text" value="Pro1"/>
ProjectName	<input type="text" value="Project1"/>
Explanation	<input type="text" value="Explanation1"/>

Save

Şekil 4.25 prCell yazılımında proje kaydı girme



Şekil 4.26 örnek bir yapay sinyal yolağı uygulaması.

M ile başlayanlar molekülleri, P ile başlayanlar proteinleri ve C ile başlayanlar da kompleks yapıları temsil etmektedir. Şekil 4.25 ve Şekil 4.26 proje kaydının yapılması ve sonrasında tasarlanmış bir sinyal yolağı gösterilmiştir.

4.2.4. Proje Import ve Export Etmek

Upload Project

ProSymbol

ProjectName

Explanation

Canvas

Save

Şekil 4.27 prCell yazılımında proje import etmek.

Başka bir kullanıcının projesini devralabilir ve devam ettirebilirsiniz. Bunun için size diğer kullanıcı projesini export edip göndermesi gerekecektir. Export edilen canvas verisi ilgili alana yüklenmelidir. Şekil 4.27'de varolan bir projenin nasıl sisteme yükleneceğini Şekil 4.28'de ise bizim projemizin başka bir kullanıcıya nasıl aktarılacağı gösterilmiştir.

Export File

```
Canvas
{"cells":[{"type":"devs.MyImageModel","size":{"width":110,"height":130},"port-body":
{"width":10,"height":10,"x":-50,"y":-5,"stroke":"gray","fill":"lightgray","magnet":"active"},"inPorts":
[],"outPorts":[],"ports":{"groups":{"in":{"position":"left","attrs":{".port-label":{"fill":"#000"},".port-body":
{"fill":"#fff","stroke":"#000","r":10,"magnet":"passive","strokeWidth":1,"width":7,"height":7,"ref-x":-6,"ref-
y":-15},"text":{"fill":"#000000"},"rect":{"fill":"#ffffff","stroke-width":1,"magnet":"passive"},"label":
{"position":"left"},"markup":"<rect class=\\"port-body\\"/>"},"out":{"position":"right","attrs":{".port-label":
{"fill":"#000"},".port-body":
{"fill":"#fff","stroke":"#00c5cd","r":10,"magnet":"passive","strokeWidth":1,"width":7,"height":7,"ref-
x":-9,"ref-y":-5},"text":{"fill":"#000000"},"rect":{"fill":"#ffffff","stroke":"#00c5cd","stroke-
width":1,"magnet":true},"label":{"position":{"name":"right","args":{"y":10,"x":3}}},"markup":"<rect
class=\\"port-body\\"/>"},"in2":{"position":"top","label":{"position":{"name":"right","args":
{"x":23,"y":-15}}},"attrs":{"text":{"fill":"#000000"},"rect":{"fill":"#ffffff","stroke-
width":1,"magnet":"passive"},".port-body":{"strokeWidth":1,"width":7,"height":7,"ref-x":10,"ref-
y":-20."maonet":"passive"}}."markup":"<rect class=\\"port-body\\"/>"},"in2":{"position":"top".label":
```

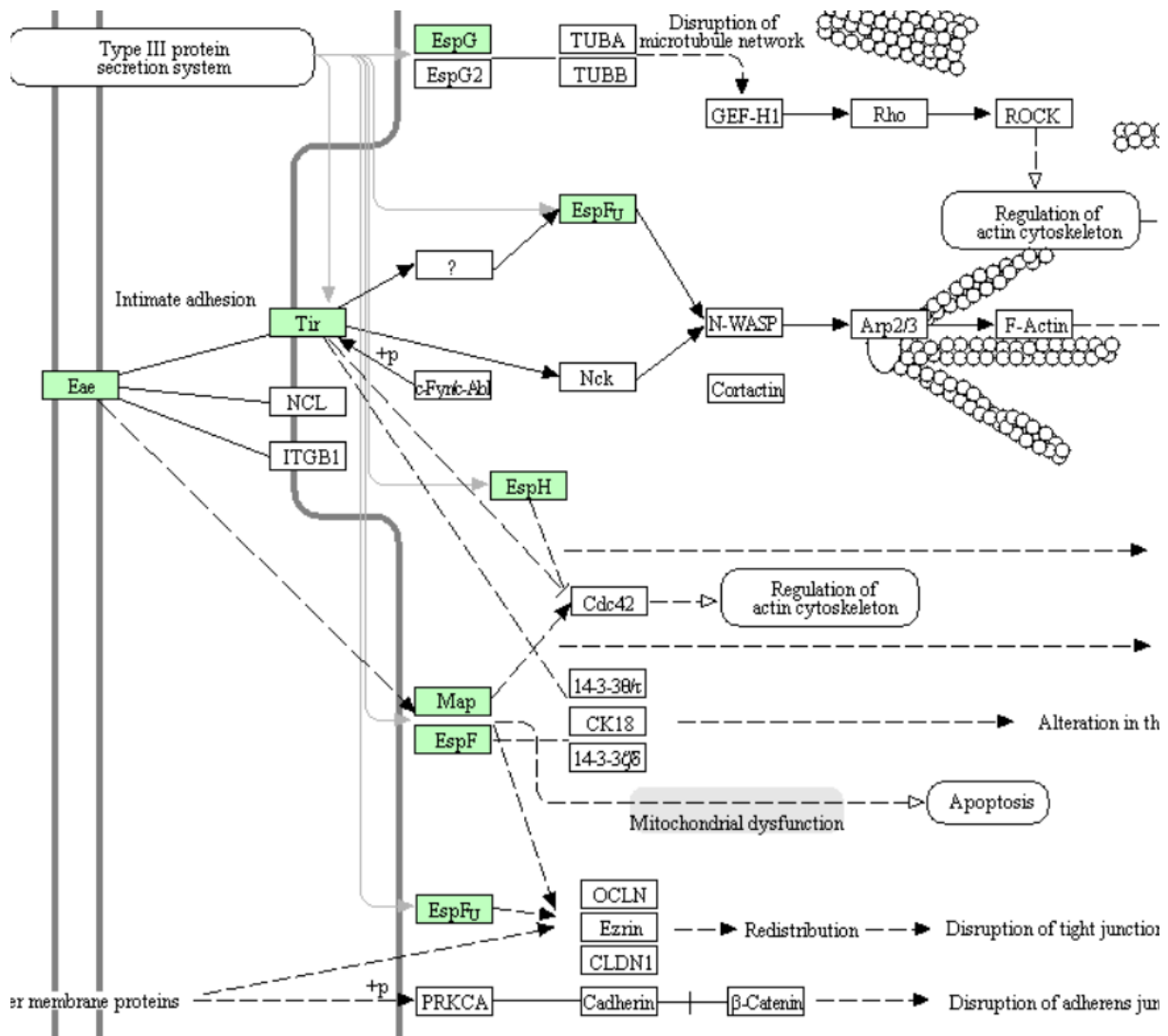
Şekil 4.28 prCell yazılımında projeyi export etmek.

4.3. Ecoli Bakterisine Ait Sinyal Yolağı Üzerine Bir Uygulama

Yaptığımız çalışmanın yapay bir örneğinden sonra şimdi de gerçek bir sinyal yolağı üzerinde denemesini yapalım. Gerçek bir sinyal yolağının prCell’de nasıl temsil edildiği, temsil edilemeyen noktaların görülmesi açısından büyük önem arz etmektedir. Keggpatway veri tabanındaki sinyal yolağının çiziminde kullanılmış olan doğruların içerdiği anlamlar burada port renkleriyle gösterilmiştir. Bu aşamada port renklerinin ne anlama geldiğini bir kez daha hatırlatmakta fayda var.

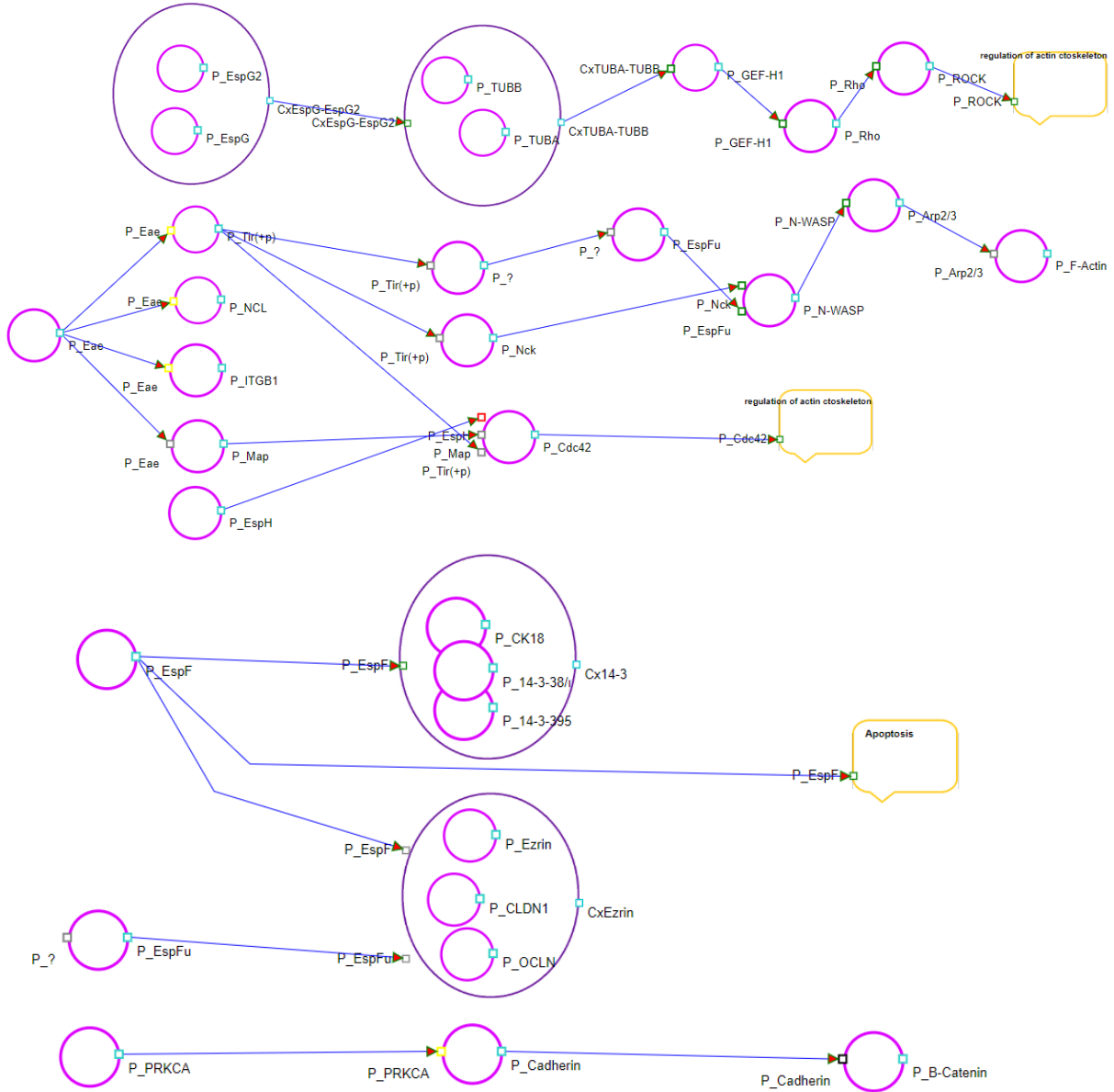
Activation	: yeşil
Inhibition	: kırmızı
Expression	: mavi
Repression	: turuncu
IndirectEffect	: gri
Statechance	: mor
Association	: sarı
Dissociation	: siyah

Keggpathway’den aldığımız örneği incelerken sinyal yolağındaki bağlantı noktalarının bazılarının belirsiz olmasından dolayı zorlandığımızı söyleyebiliriz. Ayrıca yan yana gösterilmiş proteinler(keggpathway’de bu açıdan da belirsiz durumlar vardır.) kompleks olduğu düşünüldüğünden sinyal yolağı bu duruma göre tasarlanmıştır. prCell’de bağlantılar kesindir. Eşleşme varsa bağlanır, eşleşme yoksa bağlanmaz. Bu açıdan prCell belirsizlik içermez. Şimdi keggpathwayden aldığımız sinyal yolağını prCell’de tasarlayalım.[45]



Şekil 4.29 Keggpathway'den alınmış *ecoli* bakterisine ait bir sinyal yolağı

Yukarıda görmüş olduğunuz *ecoli* bakterisine ait olan bir sinyal yolağı. Bu sinyal yolağının prCell üzerinde tekrar tasarlamak istiyoruz. Keggpathway'den alınmış olan bu sinyal yolağına baktığımızda bazı eksik yanlarının olduğu düşünülmektedir. Örneğin yana yana gelmiş proteinler bir protein kompleksini mi ifade ediyor? Eğer etmiyorsa onlara doğru çizilmiş oklar tam olarak hangisini etkiliyor anlaşılıyor. Biz bu durumu prCell'de protein kompleksi olarak yorumladık. Aynı zamanda bir proteinin modifiye olduğu gösterilmiş (Tir proteini). Bu protein aynı zamanda bir sonraki iki farklı proteine de etki ediyor ancak, bu etkisin ikisi de modifiye olduktan sonra mı yoksa biri modifiye olmadan biri modifiye olduktan sonra mı tam olarak anlaşılıyor. Biz durumda modifiye olduktan sonra etki ettiğini düşünerek prCell'de tasarladık.



Şekil 4.30 prCell de şekil 4.29 de gösterilen ecoli bakterisine ait sinyal yolağının gösterimi.

Şekil 4.29da gösterilen Keggpathway'den alınmış ecoli bakterisine ait bir sinyal yolağının prCell'de tasarımını yaparken, prCell'in Keggpathway'e göre daha net, kesin ve anlaşılır kuralları olduğuna şahit olduk.(Bknz. Şekil 4.30) Görsel olarak da daha sade bir modelleme olduğunu düşünüyoruz.Aynı zamanda bu kısımda şekli büyütme küçültme(zoom) özelliği olduğunu da belirtelim.Çalışma mantıkları ise tamamen birbirinden farklı olması açısından prCell yapay sinyal yolağı tasarlamada büyük bir fırsat sunmaktadır.

5.TARTIŞMA

Son zamanlarda hastalıkların önlenmesi ve tedavi edilmesi amacıyla hücreye dışarıdan müdahale edilmeye başlanmıştır. Hastalıkların anlaşılabilmesi ve analiz edilebilmesi için sinyal yolları önemli bir yer edinmektedir. Hücredeki olayların modellenmesi için bazı sinyal yolağı çizim programları geliştirilmiş ve halen kullanılmaktadır. Araştırmacılar hücredeki olayları incelemekte ve yeni bulgulara ulaştıkları zaman bunu, sinyal yolağı çizim programlarında modellemektedirler.

Bizim çalışmamızdaki amaçlarımızdan biri daha önce hiç var olmamış bir sinyal yolağı oluşturmaktır. Tamamıyla yapay olacak olan bu sinyal yolağı teorik olarak doğru çalışacaktır. Ancak pratikte, yani canlı hücrede bunu mümkün kılmak kolay olmayacaktır. Ancak yapay ve teorikte doğru çalışan bir sinyal yolağının canlı hücreye uygulanabilmesi çok büyük bir aşama olarak düşünülmektedir. Bu tam olarak örneğin bir bakteriyi yeniden programlamak demektir.

Bu tez çalışmasındaki diğer amaç ise bir canlıdaki bir özelliğin başka bir canlıya aktarılabilmesidir. Benzer metabolizmaların sinyal yolları karşılaştırıldığında ilgili sinyal yolağı tekrar programlanarak istenilen fonksiyonun aktarılması öngörülmektedir.

Sinyal yolağı çizmek için bazı web tabanlı uygulamalar mevcuttur. Bu uygulamalar üzerinden sinyal yolağı çizilebilir, protein ve genlerin bilgilerine bağlantılı olduğu veri tabanı üzerinden ulaşabilirsiniz. Ancak bu programlarda Ek3'te görüldüğü üzere şekilleri ve etkileşimleri seçebilirsiniz. Bu uygulamalarda sinyal yolağı çizerken özellikle bağlantılarda kurallar olmadığı için, aynı zamanda dönüşümleri gösteren yapılar tanımlanmadığı için aslında paint programından pek farkı yoktur. Sinyal yolağı programlama dili için etkileşim, dönüşüm ve bağlantıların sabit kuralları olması gerekir. Hücre içindeki olayların sayısı ve çeşidi çok fazla olduğundan dolayı etkileşim, dönüşüm ve kompleks yapıların sisteme tanımlanmasının kullanıcıya bırakılması gerekir.

Hücre programlama dili olarak karşımıza 2016 yılında MIT tarafından geliştirilmiş olan cello programlama dili çıkmaktadır. Bu hücre programlama dili verilog adındaki devre tasarım yazılımını kullanmaktadır. Elektronik devre tasarım yazılımlarında temel varlık olarak 14 tane mantık kapısı vardır. Bu mantık kapılarının kendilerine ait kuralları vardır ve tasarım yaparken bu kuralları değiştiremezsiniz. Bütün devre bu 14 kapının birbirleriyle olan bağlantıları ile tasarlanır. Tasarlanan devrenin bir akış yönü olur. Devrede kullanılan mantık kapılarının

girdileri ve çıktısı olur. İlgili kurallara göre çıktıdan elektrik çıkar veya çıkmaz. Elektrik çıkarsa bu durum "1" ile çıkmazsa "0" ile gösterilir.

Cello programlama dilinin temel mantığı verilog programındaki devre tasarım kurallarının hücredeki olaylara karşılık getirmeye çalışmasıdır. Bazı durumlara karşılık verilog programındaki bazı mantık kapılarını eşlese de bütün mantık kapıları kapsayan bir çözüm üretememiştir. Kaldı ki 14 mantık kapısının tamamına hücrede karşılık bulsa dahi hücre içindeki olayların sayısı ve çeşidi sınırlı olmadığından dolayı elektronik devre tasarım mantığının hücreye tam olarak uyarlanması mümkün görünmemektedir. Bu açıdan hücre programlama dili için farklı bir mantığa ihtiyaç duyulmuştur.

Tez çalışmamızda geliştirdiğimiz prCell programlama dilinde ise farklı ancak hücredeki olayların çoğunu modelleyebileceğiniz bir tasarım ortamı geliştirmeye çalıştık. Bileşikler, proteinler kompleks yapılar ve cello programlama dilindeki mantık kapılarına karşılık gelebilecek entity yapıları oluşturduk. Kullanıcıya özel bu programlama dilinde internet üzerinden herkes üye olabilecek ve kendine ait bir veri tabanı ve sinyal yolağı programlama imkanı bulacaktır. Hem gen ifadelerini hem de biyokimyasal reaksiyonların modellenebileceği entity yapılarını sisteme tanımlamak kullanıcıya bırakılmıştır. Bu durum hücredeki olayların çokluğu ve çeşitliliğinin karşılanabilmesi için büyük imkan sağlamaktadır. Bu dönüşümleri simgeleyen entity yapısının girdilerini, çıktılarını ve varsa kontrol kısımlarının tanımlamak kullanıcıya bırakılmıştır. Bu sayede kullanıcı sınırsız sayıda entity tanımlayabilir. Bu da hücredeki olayları temsil etme açısında büyük bir imkan sağlamaktadır.

Yakın zaman içinde www.prcell.net üzerinden yayınlamayı düşündüğümüz bu programlama dilinin kullanıcılar tarafından ilgi görmesi halinde, kullanıcıların talepleri doğrultusunda geliştirilecek ve hücredeki fonksiyonlar daha iyi temsil edilmeye çalışılacaktır. Araştırmacıların prCell'de yapacakları bir yapay sinyal yolağının canlı hücrede uygulayabilmeleri ise asıl hedefimizdir.

6.SONUÇ VE ÖNERİLER

6.1. Sonuç

Genel olarak çalışmamız yapay bir sinyal yolağı tasarlamaya imkan vermektedir. Bununla birlikte eksik olduđu bazı etkileşim ve ilişkiler de mevcut olabilir. Kullanıcıların yapması gereken sisteme üye olmak, üye olduktan sonra kütüphanelerine protein, bileşik ve entity tanımlamak olacaktır. Ne kadar çok kayıt eklenirse prCell programlama dilinin yeteneđi de o derece artacaktır. Kullanıcıların programladığı bir sinyal yolağının canlı bir hücreye uygulanabilmesi, bundan sonraki prCell çalışmalarını da önemli hale getirecektir. Günümüzde hücreye dışarıdan yapılan müdahaleler belki de bundan sonraki süreçte, önce bilgisayar ortamında kodlanacaktır. prCell programlama dili birçok açıdan bir ilk olacaktır.

6.2. Öneriler

1. prCell programlama dilinin eksik kaldığı etkileşimler ve ilişkiler modellenerek sisteme eklenmelidir.
2. Yazılım içerisinde kullanıcılar mesajlaşabilmeli ve veri paylaşımı yapabilmelidir.
3. Sinyal yolağı kodlanırken bir çıktının hangi proteinin, bileşiğın veya entity'nin girdisi olduğunun sistem içinden aranabilmesi gereklidir.
4. prCell programlama diline ait bir kullanım klavuzu hazırlanabilir.
5. İlgi görmesi durumunda kullanıcıların talepleri takip edilmeli ve prCell yeni ihtiyaçlara göre güncellenebilmelidir.
6. İnternet üzerinden yayınlanmadan önce testlerinin yapılması ve farklı görüş ve tavsiyelerin dikkate alınması büyük önem arz etmektedir.
7. Elde etmek istediğimiz bir proteinin nasıl elde edilebileceğine yönelik olarak geriye doğru sinyal yolağının program tarafından çizilmesi de amaçlanmaktadır.

7. KAYNAKLAR

1. Bradshaw Ralph A, Dennis Edward A, eds. Handbook of Cell Signaling. 2nd ed. Amsterdam, Netherlands: Academic Press; 2010.
2. Geoffrey M Cooper. The Cell: A Molecular Approach. 2nd edition Affiliations Boston University Sunderland (MA): Sinauer Associates; 2000. ISBN-10: 0-87893-106-6
3. Marianna A. Zolotovskaia, Maxim I. Sorokin, Sergey A. Roumiantsev, Nikolay M. Borisov and Anton A. Buzdin. Pathway Instability Is an Effective New Mutation-Based Type of Cancer Biomarkers. *Front Oncol.* 2018; 8: 658.
4. Patricia Sebastian-Leon, Enrique Vidal, Pablo Minguez, Ana Conesa, Sonia Tarazona, Alicia Amadoz, Carmen Armero, Francisco Salavert, Antonio Vidal-Puig, David Montaner, and Joaquín Dopazo. Understanding Disease Mechanisms With Models Of Signaling Pathway Activities. *BMC Syst Biol.* 2014; 8: 121.
5. Stryer L, Berg JM, Tymoczko JL . Biochemistry (5th ed.). San Francisco: W.H. Freeman; 2002.
6. Ulrike Wittig, Andreas Weidemann, Renate Kania, Christian Peiss and Isabel Rojas. Classification of Chemical Compounds to Support Complex Queries in a Pathway Database. *Comp Funct Genomics*;2004 ; 5(2):156-62.
7. Lodish H, Berk A, Matsudaira P, Kaiser CA, Krieger M, Scott MP, Zipurksy SL, Darnell J Molecular Cell Biology (5th ed.).New York, New York: WH Freeman and Company.2004
8. EBI External Services [Internet] 2010. The Catalytic Site Atlas at The European Bioinformatics Institute.[Erişim Tarihi 12 Nisan 2019]. Erişim adresi: <https://www.ebi.ac.uk/thornton-srv/databases/CSA/>
9. Philip E. Bourne. Life is three-dimensional, and it begins with molecules. *PLoS Biol.* 2017 Mar; 15(3): e2002041.
10. Boutureira O, Boutureira O1, Bernardes GJ. Advances in chemical protein modification, *Chem Rev.* 2015 Mar 11;115(5):2174-95.
11. Marsh JA, Teichmann SA. Annu Structure, dynamics, assembly, and evolution of protein complexes, *Annu Rev Biochem.* 2015;84:551-75.
12. Xiao-Li Li, Chuan-Sheng Foo, See-Kiong Ng. Discovering Protein Complex in Dense Reliable Neighborhoods Of Protein Interaction Networks. *Comput Syst Bioinformatics Conf.* 2007;6:157-68.
13. Jose B Pereira-Leal, Emmanuel D Levy, and Sarah A Teichmann "The origins and evolution of functional modules: lessons from protein complexes". *Philos Trans R Soc Lond B Biol Sci.* 2006 Mar 29; 361(1467): 507–517
14. National Research Council (US) Committee on Research. Opportunities in Biology. Washington (DC): National Academies Press (US); 1989.
15. Price NC, Stevens L, X.Hartwell LH, Hopfield JJ, Leibler S, Murray AW . Fundamentals of enzymology: The cell and molecular biology of catalytic protein. Oxford, New York: Oxford University Press; December 1999.

16. Pereira-Leal JB, Levy ED, Teichmann SA .From molecular to modular cell biology. Nature. March 2006; 402: 6761
17. Hansen TM, Baranov PV, Ivanov IP, Gesteland RF, Atkins JF. Maintenance of the correct open reading frame by the ribosome. EMBO Reports.2003; **4** (5): 499–504.
18. Berk V, Cate JH . Insights into protein biosynthesis from structures of bacterial ribosomes. Current Opinion in Structural Biology.2007; **17** (3): 302–9.
19. Robinson PK.Enzymes: Principles and Biotechnological Applications.Essays Biochem. 2015;59:1-41.
20. Prof. A K Rai and Prof. R P Sinha. Gene : Expression and Regulation (in Recent Advances in Life Sciences).1st ed. New Delhi, India, IK international publisher.2010;353-381
21. Daniel Medina-Cleghorn and Daniel K. Nomura Chem Biol.Exploring Metabolic Pathways and Regulation through Functional Chemoproteomic and Metabolomic Platforms Author manuscript.Chem Biol. 2014 Sep 18; 21(9): 1171–1184.
22. Lu LJ, Sboner A, Huang YJ, Lu HX, Gianoulis TA, Yip KY, Kim PM, Montelione GT, Gerstein MB.Comparing classical pathways and modern networks: towards the development of an edge ontology.Trends Biochem Sci. 2007 Jul;32(7):320-31.
23. Hojung Nam, Tom Conrad and Nathan E. Lewis. Metabolic pathway evolution and cellular objectives. Curr Opin Biotechnol. 2011 Aug 1; 22(4): 595–600.
24. Cell Illustrator [İnternet]. 2009 [Erişim Tarihi 12 Mart 2019]. Erişim adresi: <http://www.cellillustrator.com>
25. Celldesigner [İnternet]. 2004 [Erişim Tarihi 12 Mart 2019]. Erişim adresi: <http://www.celldesigner.org>
26. Pathvisio [İnternet]. 2008 [Erişim Tarihi 12 Mart 2019]. Erişim adresi: <http://www.pathvisio.org/>
27. Reactome [İnternet]. 2009 [Erişim Tarihi 12 Mart 2019]. Erişim adresi: <http://www.reactome.org>
28. Wikipathways [İnternet]. 2008 [Erişim Tarihi 12 Mart 2019]. Erişim adresi: <http://wikipathways.org/index.php/WikiPathways>
29. Nielsen AA, Der BS, Shin J, Vaidyanathan P, Paralanov V, Christopher A. Voigt ve arkadaşları. Genetic circuit design automation. Science. 2016 Apr 1;352(6281):aac7341
30. Endy, D. Foundations for engineering biology. Nature.2005; 438, 449–453.
31. Osbourn AE, O'Maille PE, Rosser SJ, Lindsey K . Synthetic biology. 4th New Phytologist Workshop, Bristol, UK, June 2012. The New Phytologist.2012; **196** (3): 671–7.
32. Schwille P. Bottom-up Synthetic Biology: Engineering in a Tinkerer's World. Science. 2011;**333**(6047):12524.
33. Jennifer A.N. Brophy and Christopher A. Voigt. Principles of Genetic Circuit Design Author manuscript. Nat Methods. 2014 May; 11(5): 508– 520.
34. Genome [İnternet]. 1995 [Erişim Tarihi 12 Mayıs 2019]. Erişim adresi:

https://www.genome.jp/kegg/document/help_pathway.html

35. Pathwaycommons [İnternet]. 2011 [Eriřim Tarihi 12 Mayıs 2019]. Eriřim adresi:
https://www.pathwaycommons.org/pc/sif_interaction_rules.do
36. Brian Knight, Devin Knight, Jessica M. Moss, Mike Davis, Chris Rock. Professional Microsoft SQL Server 2014 Integration Services. Indianapolis, IN, United States of America: John Wiley & Sons, Inc;2014.
37. Adam Freeman , Pro Asp.net mvc5, 5th Ed. Apress; 2013 ISBN-13: 978-1430265412 ISBN-10: 143026541
38. Jose Rolando Guay Paz. Beginning asp.net mvc4, Apress Media, LLC is a California LLC: Apress Media;2013.
39. Jointjs [İnternet]. 2009 [Eriřim Tarihi 17 řubat 2018]. Eriřim adresi:
<https://www.jointjs.com/>
40. Custom Elements, [İnternet]. 2009 [Eriřim Tarihi 17 řubat 2018]. Eriřim adresi:
<https://resources.jointjs.com/tutorial/custom-elements>
41. Links, [İnternet]. 2009 [Eriřim Tarihi 17 řubat 2018]. Eriřim adresi:
<https://resources.jointjs.com/tutorial/links>
42. Link Tools, [İnternet]. 2009 [Eriřim Tarihi 17 řubat 2018]. Eriřim adresi:
<https://resources.jointjs.com/tutorial/link-tools>
43. Ports, [İnternet]. 2009 [Eriřim Tarihi 17 řubat 2018]. Eriřim adresi:
<https://resources.jointjs.com/tutorial/ports>
44. Connecting-by-dropping, [İnternet]. 2009 [Eriřim Tarihi 17 řubat 2018]. Eriřim adresi:
<https://resources.jointjs.com/tutorial/connecting-by-dropping>
45. Keggpathway Exp., [İnternet]. 2009 [Eriřim Tarihi 19 Temmuz 2019]. Eriřim adresi:
https://www.genome.jp/kegg-bin/show_pathway?ece05130+M00542

8.EKLER

EK-1: Etkileşim Kuralları

https://www.pathwaycommons.org/pc/sif_interaction_rules.do adresinden alınmıştır.

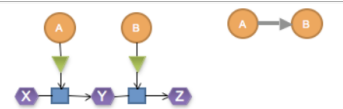
METABOLIC_CATALYSIS: The first entity catalyzes a reaction that either consumes or produces the second entity. This interaction is directed.



REACTS_WITH: The entities participate in a conversion as substrates or products. Controllers are not included. This interaction is undirected.



SEQUENTIAL_CATALYSIS: The entities catalyze two conversions that are connected via a common molecule, e.g. the first entity produces a substrate that is consumed by the second entity. This interaction is directed.



STATE_CHANGE: The first entity controls a reaction that changes the state of the second entity, e.g. by phosphorylation or other posttranslational modification, or by a change in subcellular location. This interaction is directed.









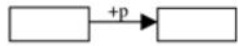

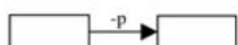
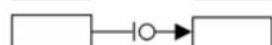
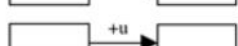
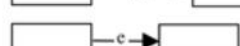
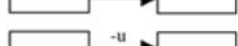

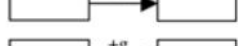
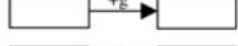
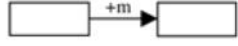




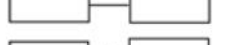



Legend

Protein	Conversion	Molecular Complex
Modified Protein	Activation	
Small Molecule	Inhibition	

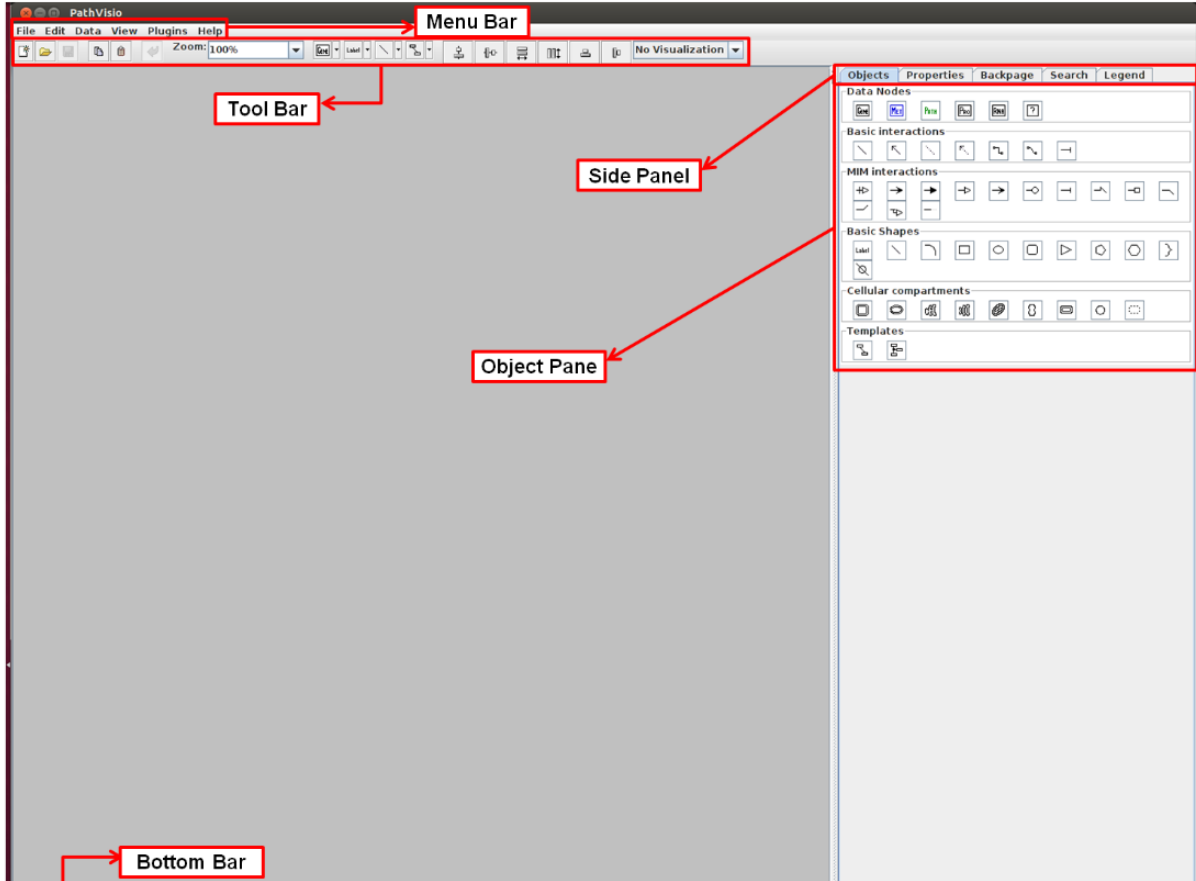
EK-2: KEGG Pathway Map

https://www.genome.jp/kegg/document/help_pathway.html adresinden alınmıştır.

Notation		Each pathway is identified by a five-digit number preceded by one of: map, ko, ec, rn, and three- or four-letter organism code. The pathway map is drawn and updated with the notation shown below.	
Objects		gene product, mostly protein but including RNA	
		chemical compound, DNA and other molecule	
		map	
		Arrows	
			molecular interaction or relation
			link to/from another map
			indirect link or unknown reaction
			missing interaction (eg., by mutation)
			drug structure link or pointer used to add legend
Protein-protein interactions		Gene expression relations	
	phosphorylation		expression
	dephosphorylation		repression
	ubiquitination		expression
	deubiquitination		repression
	glycosylation		
	methylation		
	activation		
	inhibition		
	indirect effect or state change		
	binding / association		
	dissociation		
	complex		
		Enzyme-enzyme relations	
			two successive reaction steps

EK-3: Pathvisio Sinyal Yolağı Çizim Ekranı.

<https://www.pathvisio.org/documentation/> adresinden alınmıştır.



EK-4: Dijital Makbuz**Dijital Makbuz**

Bu makbuz ödevinizin Turnitin'e ulaştığını bildirmektedir. Gönderiminize dair bilgiler şöyledir:

Gönderinizin ilk sayfası aşağıda gönderilmektedir.

Gönderen: Mecit Aktaş
Ödev başlığı: prCell: SENTETİK BİYOLOJİ İÇİN S...
Gönderi Başlığı: prCell: SENTETİK BİYOLOJİ İÇİN S...
Dosya adı: Mecit Aktaş.pdf
Dosya boyutu: 3.36M
Sayfa sayısı: 93
Kelime sayısı: 13,444
Karakter sayısı: 89,069
Gönderim Tarihi: 30-Eyl-2019 01:35PM (UTC+0300)
Gönderim Numarası: 1182993602



prCell: SENTETİK BİYOLOJİ İÇİN SİNYAL YOLAĞI
PROGRAMLAMAYA YÖNELİK BİR YAZILIM

ORJİNALLİK RAPORU

% 7	% 6	% 0	% 7
BENZERLİK ENDEKSİ	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCİ ÖDEVLERİ

BİRİNCİL KAYNAKLAR

1	Submitted to Hacettepe University Öğrenci Ödevi	%5
2	www.openaccess.hacettepe.edu.tr:8080 İnternet Kaynağı	%1
3	openaccess.hacettepe.edu.tr:8080 İnternet Kaynağı	<%1
4	Submitted to Trakya University Öğrenci Ödevi	<%1
5	K. L. Mueller. "Dialing down allergic responses", Science, 2016 Yayın	<%1
6	Submitted to Ataturk Universitesi Öğrenci Ödevi	<%1
7	Submitted to Yeditepe University Öğrenci Ödevi	<%1
8	repository.uinjkt.ac.id İnternet Kaynağı	<%1

9. ÖZGEÇMİŞ

KİŞİSEL BİLGİLER:

Adı ve Soyadı :Mecit Aktaş
Doğum Yeri ve Tarihi : Amasya 1984
Uyruđu : T.C.
İletişim Adresi ve Telefonu : T.C. Aile, Çalışma ve Sosyal Hizmetler Bakanlığı B.İ.D.B.
Emek Mahallesi 17. Cadde No:13 Pk: 06520 Emek / ANKARA
0553 374 2550

ÖĞRENİM DURUMU:

Lisans : Ege Üniversitesi Matematik
Yüksek Lisans : Hacettepe Üniversitesi Biyoinformatik Yüksek Lisans Programı

MESLEKİ DENEYİMLERİ:

2015-Halen T.C. Aile, Çalışma ve Sosyal Hizmetler Bakanlığı Programcı

GÖREV ALDIĞI PROJELER:

1. Vakıf Personel Sistemi
2. Korunmaya Muhtaç Çocuklar Bilgi Sistemi
3. Şehit Gazi Bilgi Sistemi