YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DAĞITIK DVM KULLANARAK MIRNA HEDEF GEN TAHMİNİ YAPILMASI

Bilgisayar Müh. Niyazi ELVAN

FBE Bilgisayar Mühendisliği Anabilim Dalında Hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Yrd. Doç. Dr. A. Gökhan YAVUZ

İSTANBUL, 2009

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ

$\mathbb{N}$ Nükleotid dizisi
$\mathfrak{M}$ Eşleşme dizisi

KISALTMA LİSTESİ

DNA	Deoksiribo Nükleik Asit
RNA	Ribo Nükleik Asit
miRNA	mikro Ribo Nükleik Asit
tRNA	taşıyıcı Ribo Nükleik Asit
mRNA	mesajcı Ribo Nükleik Asit
GTP	Guanozin Tri Phosfat
DVM	Destek Vektör Makinesi
RBF	Radial Basis Function
AHB	Ana Hesaplama Birimi
UHB	Uç Hesaplama Birimi
MFE	Minimum Free Energy

ŞEKİL LİSTESİ

ÇİZELGE LİSTESİ

ÖNSÖZ

Bu tez çalışmasını yaparken elinden gelen hiçbir gayreti esirgemeyen ve beni hem bu tez çalışması sırasında hem de Yıldız Teknik Üniversitesi'ndeki lisans ve yüksek lisans öğrenimim boyunca sürekli destekleyen değerli danışmanım A. Gökhan Yavuz'a teşekkür etmeyi bir borç bilirim.

Niyazi Elvan
Ağustos, 2009

ÖZET

DAĞITIK DVM KULLANARAK miRNA HEDEF GEN TAHMİNİ

Niyazi Elvan
Bilgisayar Mühendisliği, Yüksek Lisans Tezi

Son yıllarda mikrobiyoloji alanındaki gelişmeler bu alana bilgisayar bilimlerinin desteğinin giderek artması ile ivme kazanmıştır. Mikrobiyolojinin en etkili alt dallarından biri de Genetik bilimidir. Genetik özellikle günümüzde üzerinde en çok çalışma yapılan alanlardan biridir. Gen ve Genom araştırmaları, proteinlerin üç boyutlu yapılarının tespiti, DNA dizilimlerin keşfi gibi gelişmeler bu konu üzerinde çalışan araştırmacıları heyecanlandıran gelişmeler olarak nitelendirilebilir. Zira yapılan her çalışma yeni bir çalışma alanı doğurmakla kalmayıp günlük hayatta çok hızlı uygulanabilen çözümlere dönüştürülebilmektedir.

Yapılan araştırmaların ve ortaya çıkan ihtiyaçlarının bir sonucu olarak bilgisayar bilimleri, mikrobiyoloji ve istatistik bilimlerinin bir bileşkesi olan Biyoenformatik bilimi türemiştir. Biyoenformatik bu üç bilim dalını çok iyi bir şekilde yorumlayarak ortaya şaşırtıcı sonuçlar koyabilmektedir. DNA ve RNA dizilerinin keşfinin ardından, bunlara ait dizilimlerin elde edilmesi ve akabinde bu dizilimlerin istatistik ve bilgisayar bilimleri yöntemleri ışığında değerlendirilmesi biyoenformatiğin ilgilendiği konuların başında gelir. Bu konularla ilgili çalışmalar sürerken biyolojik gelişmeler de olmuştur. Varlığı ilk kez 1993 yılında keşfedilen ve ancak 2001'de ismi konulan miRNA molekülleri bugün Biyoenformatik alanında çalışan birçok araştırmacının gözdesi olmuştur. Başlangıçta hücre içindeki işlevinin tam olarak ne olduğu bilinmezken bugün belirli miRNA türlerinin hücrede ne işleve sahip olduğu bilinmektedir.

miRNA hakkındaki çalışmalar genel olarak iki ayrı dalda ilerlemektedir. Birinci alan miRNA'ların kalıtsal olarak iletilip iletilemediğini, hangi süreçlerden etkilendiğini ve bu moleküllerin nasıl oluştuğunu araştırır. Diğer alan ise miRNA'ların hangi süreçlere, nasıl etki ettiğini ve hedef aldığı genleri araştırır. Bu çalışmada miRNA'ların hedef aldığı genlerin bilinen örneklerden yola çıkılarak makine öğrenmesi yöntemlerinden DVM kullanılarak tahmin edilmesi hedeflenmiştir. Üç ana parçadan oluşan sistem, ilk aşamada DVM için veri toplanması ve bu verilerden özellik kümesi dikkate alınarak özellik çıkarılmasını sağlar. İkinci aşamada oluşturulan özelliklerin öğrenme verisi olarak kullanılması ve DVM'nin eğitilmesi söz konusudur. Son aşama ise DVM'in sınanması ve sorgulanabilir şekilde hazır duruma getirilmesidir.

Önceki uygulamalar incelendiğinde DVM'lerin işlemci zamanı düşünüldüğünde yüksek başarım gerektirdiği bilinmektedir. Bu çalışmayı oluşturan üç ana parça mpich kütüphanesi kullanılarak paralel ortamda geliştirilmiştir. Uygulamaların tek işlemci üzerinde çalışan diğer uygulamalara göre yüksek oranda başarım elde ettiği görülmüştür. Bu sayede çok büyük veri kümelerinin işlenmesi için harcanacak sürenin de büyük oranda azalacağı öngörülmektedir.

Anahtar Kelimeler: Destek Vektör Makinesi, Paralel Hesaplama, miRNA hedef gen tahmini, biyoenformatik

JÜRİ:

1. Yrd. Doç. Dr. A. Gökhan YAVUZ (Danışman) Tarih : 31.08.2009
2. Prof. Dr. A. Coşkun SÖNMEZ Sayfa sayısı : 48
3. Yrd. Doç. Dr. Lale ÖZYILMAZ

ABSTRACT

miRNA TARGET GENE PREDICTION USING DISTRIBUTED SVM

Niyazi Elvan
Computer Engineering, M.S. Thesis

In recent years, the improvement of Microbiology has been enhanced by the help of Computer Sciences. Genetics, one of the main branches of Microbiology, has emerged as a challenging area of research. Improvements such as prediction of protein 3D stuctures, Gene/Genome expressions and DNA sequence analysis has arised with the interest of many researchers.

Bioinformatics is the application of information technology to the field of molecular biology. The term bioinformatics was coined by Paulien Hogeweg in 1979 for the study of informatic processes in biotic systems. Bioinformatics now entails the creation and advancement of databases, algorithms, computational and statistical techniques, and theory to solve formal and practical problems arising from the management and analysis of biological data. Over the past few decades rapid developments in genomic and other molecular research technologies and developments in information technologies have combined to produce a tremendous amount of information related to molecular biology. It is the name given to these mathematical and computing approaches used to glean understanding of biological processes. Common activities in bioinformatics include mapping and analyzing DNA and protein sequences, aligning different DNA and protein sequences to compare them and creating and viewing 3-D models of protein structures.

MicroRNA (miRNA) is a small non-coding RNA which plays a significant role in plants and animals as a regulator in gene expression. A mature miRNA generally binds to the 3' site of its target gene which is usually complementary to binding site of miRNA. There are already computational methods proposed for predicting miRNA targets. Most of these methods focused on biological aspects of the problem and are lacking computational state-of-the-art.

This thesis work proposes a method used for fastening the feature extraction from miRNA&Target Gene sequence data for support vector machine (SVM) taking the advantage of parallel programming. A parallel algorithm has been implemented that takes the sequence data of miRNA::Target Gene pairs as input and calculates the structural, thermo-dynamical and position-based features for SVM as output. Using the data partitioning technique, each computing node performs the calculations on sequence files pointed by the main node and sends the result to it. We have measured the scalability of our algorithm on a parallel cluster environment. The more computing nodes exist, the less computing time is spent.

Keywords: Support Vector Machines, Parallel Computing, Bioinformatics, miRNA target prediction

JURY:

1. Assist. Prof. A. Gökhan YAVUZ (Supervisor) Date : 31.08.2009
2. Prof. A. Coşkun SÖNMEZ Page :48
3. Assist. Prof. Lale ÖZYILMAZ

1. GİRİŞ

Genetik özellikle günümüzde üzerinde en çok çalışma yapılan alanlardan biridir. Gen ve Genom araştırmaları, protein üç boyutlu yapılarının tespiti, DNA dizilimlerin keşfi gibi gelişmeler bu konu üzerinde çalışan araştırmacıları heyecanlandıran gelişmeler olarak nitelendirilebilir. DNA ve RNA dizilerinin keşfinin ardından, bunlara ait dizilimlerin elde edilmesi ve akabinde bu dizilimlerin istatistik ve bilgisayar bilimleri yöntemleri ışığında değerlendirilmesi bilimsel açıdan yeni bir ufuk yaratmıştır. Bu konularla ilgili çalışmalar sürerken biyolojik tarafta da gelişmeler de olmuştur. Varlığı ilk kez 1993 yılında keşfedilen ve ancak 2001'de ismi konulan miRNA molekülleri biyoenformatik alanında çalışan birçok araştırmacının gözdesi olmuştur. Başlangıçta hücre içindeki işlevinin tam olarak ne olduğu bilinmezken bugün belirli miRNA türlerinin hücrede hangi işleve sahip olduğu bilinmektedir.

Bundan sonraki bölümde protein sentezlenme süreci, tRNA ve mRNA'nın biyolojik etkileşimi ve miRNA'nın oluşumu hakkında bilgi verilmiştir. Üçüncü bölümde makine öğrenmesinin ne olduğu, kullanılan güncel yöntemlerin nasıl olduğu incelenmiştir. Dördüncü bölümde bu tezin temelini oluşturan problem ortaya konmuş ve bu probleme sunulan çözüm anlatılmıştır. Buna göre bir miRNA'nın hedef aldığı genleri tahmin etmek için kullanılan DVM yönteminin başarımı incelenmiş, DVM'nin özellik çıkarma aşamasında sıkıntılar olduğu tespit edilmiştir. Artan miRNA sayısı özelliklerin hesaplanması için gereken zamanın artmasına sebep olmaktadır. Bu tezde geliştirilen paralel hesaplama yöntemi ile özellik hesaplama için harcanan zamanın azalması sağlanmıştır. Geliştirilen yöntem dördüncü bölümde ayrıntılı olarak anlatılmıştır. Sonraki bölümde ise geliştirilen yöntemin farklı veri kümeleri üzerinde sınanması ile elde edilen deneysel sonuçlar ve bu sonuçların karşılaştırmalı yorumları verilmiştir.

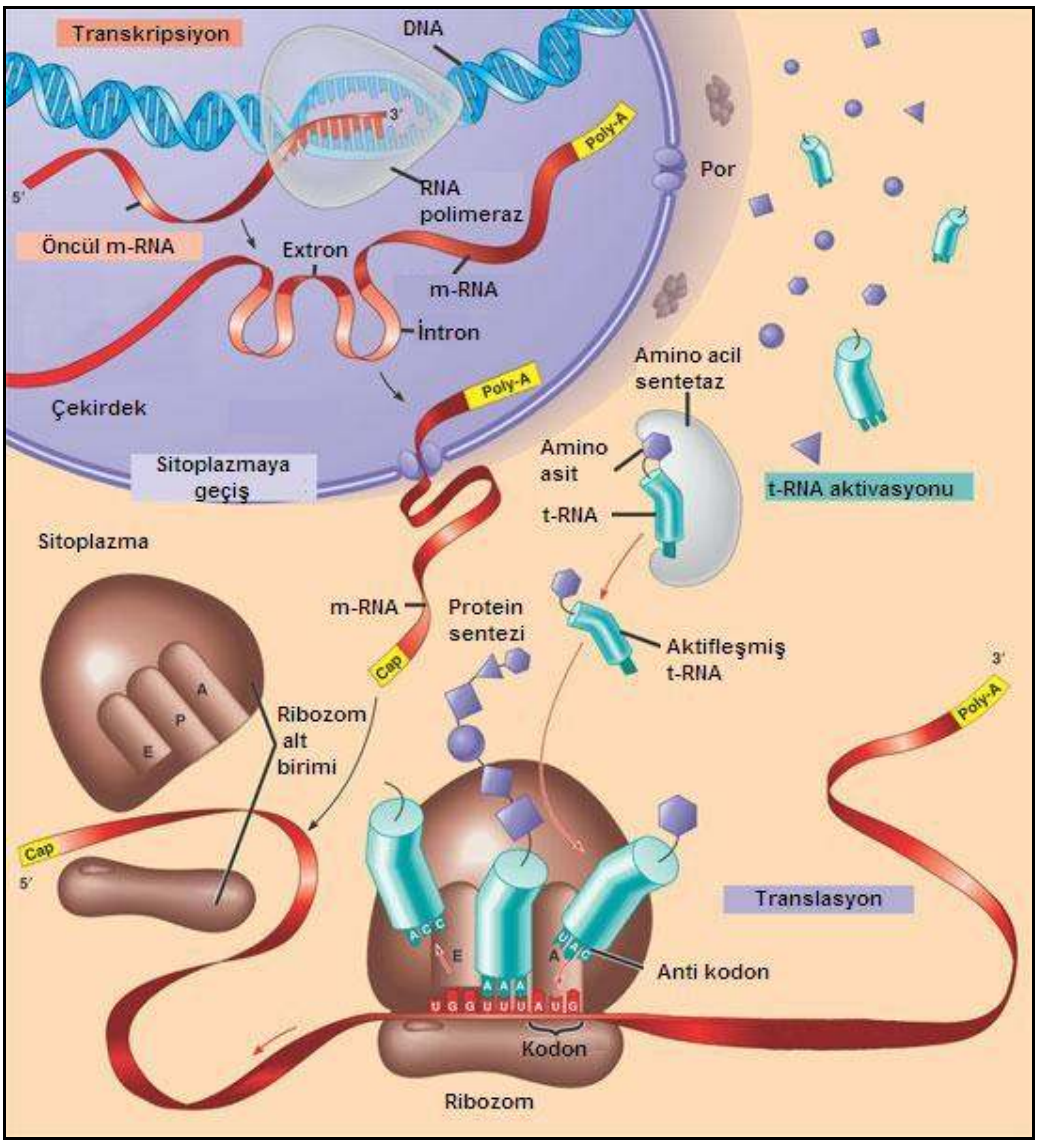
2. MIRNA OLUŞUMU VE YAPISI

Biyoloji gözleme dayalı bir bilim dalı olarak varlığını sürdüregelmiştir. Her ne kadar güncel gelişmeler bu kuralı bozamadıysa da günümüzde oluşan biyolojik veri miktarı ciddi anlamda artmıştır. Artan veri miktarı beraberinde bu verilerin işlenmesi ve bilgiye dönüştürülmesi gerekliliğini de ortaya çıkarmıştır. Bunun yanında oluşan verilerin saklanması, düzenlenmesi ve paylaşıma sunulması için de özel yöntemler ve araçlar geliştirilmesi gerekmiştir. Biyoenformatik, geniş anlamda biyolojik verilerin anlamlı bilgiye dönüştürülmesi sürecinde kullanılan yöntemler ve bu yöntemlerin bilgisayar teknolojileri kullanılarak gerçeklenmesini kapsayan disiplinlerarası bir bilim dalı olarak nitelendirilebilir.

1960'larda başlayan bilgisayar uygulamalarının biyolojide kullanılması girişimi, her iki alandaki teknolojik gelişime paralel olarak hızla ilerlemiş ve böylelikle ortaya çıkan biyoenformatik dalı bugün en gözde akademik ve endüstriyel sektörlerin başına geçmiştir. Bilgisayarların moleküler biyolojide kullanımı üç boyutlu moleküler yapıların grafik temsili, moleküler dizilimler ve üç boyutlu moleküler yapı veritabanları oluşturulması ile başlamıştır. Kısa sürede çok yüksek miktarda veri üreten, gen ekspresyonu, protein-protein ilişkisi, biyolojik olarak aktif molekül araştırmaları, bakteri, maya, hayvan ve insan genom projeleri gibi biyolojik deneylerin doğurduğu talep sonucunda, bu alandaki bilişim uygulamaları neredeyse takip edilemez bir hızda gelişmiştir. Biyoenformatik dalının ayrı bir disiplinlerarası bilim dalı olarak tanınması da son 10 yılda gerçekleşmiştir.

Biyoenformatiğin ilgi alanına giren ana konular şu şekilde sıralanabilir :

1. **DNA Dizi Çözümlemesi** : Bir DNA molekülündeki nükleotid (Adenine, Guanine, Cytosine, Thymine) dizilimine ait sırayı bulmayı sağlayan yöntemler bütünüdür. Nükleotidlerin hangi sırada dizili olduğu kalıtım ve evrimsel gelişim süreci açısından önem taşımaktadır. Dizi çözümlemesi canlı türleri arasındaki genetik benzerliklerin bulunması, tür içinde kalıtım ile iletilen özelliklerin belirlenmesi, genetik hastalıkların teşhisi gibi konularda temel seviyede yararlı bilgiler sağlayabilmektedir.
2. **Genom Bulgulama :** Bir DNA dizisinde yer alan genlerin ve diğer biyolojik özniteliklerin bulunması ve işaretlenmesidir. Genlerin işlevleri hakkında araştırma yapmak istendiğinde ilk adımı teşkil eder.
3. **Hesaplamalı Evrimsel Biyoloji :** Türlerin soyları ve zaman içinde gelişimlerini inceleyen çalışmalardır. Bilgisayar teknolojisi bu alanda yapılan araştırmalara çok büyük faydalar sağlamıştır.



Şekil 2-1 Protein biyosentezi

4. **Gen Ekspresyonu :** Bir gendeki bilginin kullanılıp işlevsel bir ürün sentezlenmesi sürecine verilen isimdir. Bu süreç sonunda oluşan ürün genellikle bir proteindir. Ancak protein oluşumunu sağlamayan genlerde oluşan ürünler mesajcı RNA (mRNA) veya taşıyıcı RNA (tRNA) olabilmektedir. Gen ekspresyon süreci transkripsiyon, translasyon ve translasyon sonrası işlemler olarak farklı adımlarda incelenebilir.

   Aslında genler DNA'daki belirli bir bölgede yer alan ve kendi başına etkin işlevi olmayan sarmal yapılardır. Bir geni dolaylı yoldan etkin kılan şey genin sahip olduğu aminoasit dizilim sırasıdır. Bu dizilim DNA'dan mRNA'ya aktarılır. Bu işleme transkripsiyon denir. Transkripsiyon hücre çekirdeğinde gerçekleşen bir olaydır. Bu adımdan sonra oluşan mRNA hücre çekirdeğini terk ederek sitoplazmaya ulaşır. Bu

noktada sitoplazmada boş halde bulunan bir ribozoma bağlanan mRNA protein sentezini başlatır. mRNA üzerinde taşıdığı bilgi ile kendisini kopyaladığı genin kodladığı proteinlerin sentezlenmesini sağlar. Şekil 2-1'de protein sentezinin aşamaları ayrıntılı bir biçimde görülmektedir.

5. **Biyolojik Sistemlerin Modellenmesi :** Sistem biyolojisi olarak da adlandırılan bu alan, biyolojik sistemlerin ve alt sistemlerin (metabolitik ağlar, hücresel tepkime zincirleri, gen düzenleme ağları, vs.) bilgisayar ortamında simulasyonunu gerçekleştirebilmek için yapılan analizleri ve görsel tasarım öğelerini kapsar. Örneğin bu çalışmalar sayesinde insandaki bağışıklık sistemi bilgisayar ortamında simüle edilebilir**,** oluşturulacak biyolojik sistem ağıyla da yeni ortaya çıkan bir hastalığın belirli bir toplulukta nasıl etki edeceği tahmin edilebilir.

6. **Karşılaştırmalı Gen Bilimi :** Karşılaştırmalı genom analizinin temeli ortolog genler diye bilinen aynı atadan gelen farklı organizmalarda aynı özellikleri temsil eden genler arasındaki ilişkileri ortaya koyar. Dolayısıyla türlerin ortaya çıkışı, türlerin sınıflandırılması ve mutasyonlar bu disiplin altında incelenmektedir.

7. **Protein Yapılarının Tahmini :** Proteinlerin üç boyutlu yapılarının tahmin edilmesine yönelik çalışmalar günümüzde yoğun bir şekilde devam etmektedir. Şimdiye kadar yapılan çalışmaların büyük bir bölümünde kullanılan yöntemler buluşsal oldukları için hiçbir zaman en doğru çözümü önerdikleri kuramsal olarak ispat edilememiştir. İspatlar, yapılan tahminlerin ardından proteinin X-ray resmi çekildikten sonra elde edilen sonuçlar ile karşılaştırılarak yapılmıştır.

   Bir proteine ait üç boyutlu yapının tahmin edilebilmesi için o proteinin birincil yapı diye adlandırılan aminoasit dizilimine ihtiyaç duyulmaktadır. Birincil yapı en temel ifade şekli olmakla birlikte proteinler üzerinde yapılan araştırmalarda oldukça önemli bir yer teşkil eder. Birincil yapıdan belirli yöntemler kullanılarak elde edilen ikincil yapı proteine ait ilk üç boyutlu tahmindir. Bu yapı proteinin kabaca nasıl göründüğünü gösterir ancak atomların hangi durumlarda ne şekilde bulunduğunu göstermez. Üçüncül yapı adıyla geçen yapıda ise bir proteinin moleküler seviyede uzayda nasıl görüneceğini gösteren ifadeler mevcuttur.

   Proteinlerin üç boyutlu yapılarının tahmin edilebilmesi için yapılan çalışmalar sırasında ilginç deneyimler yaşanmıştır. Örneğin, birincil yapı verileri incelenerek aralarında çok benzerlik olmadığı düşünülen ancak üç boyutlu yapıları (ikincil ve üçüncül yapılar) incelendiğinde birbirine benzer özellik gösteren ve işlevsel olarak da

benzerliğe sahip proteinlerin varlığı keşfedilmiştir. İşte bu yüzden proteinlerin üç boyutlu yapılarının doğru bir biçimde tahmin edilebilmesi önem taşımaktadır. Günümüzde üzerinde yoğun şekilde çalışmalar yapılan başlıca konulardandır. Hatta, IBM firmasının ünlü süperbilgisayar projesi BlueGene üç boyutlu yapı tahminini sezgisel olmadan, tüm ihtimalleri göz önünde bulundurarak gerçekleştirmek üzere geliştirilmiş bir projedir.

2.1 Protein Biyosentezi

Protein biyosentezi, hücrenin protein sentezlemesi için gereken bir biyokimyasal süreçtir. Bu terim bazen sadece protein translasyonu anlamında kullanılsa da transkripsiyon ile başlayıp translasyonla biten çok aşamalı bir süreçtir. Prokaryotlarda ve ökaryotlarda ribozom yapısı ve yardımcı proteinler bakımından farklılık göstermesine karşın, temel mekanizma korunmuştur. Bu sürecin genel hata oranı 10^{-4} civarındadır.

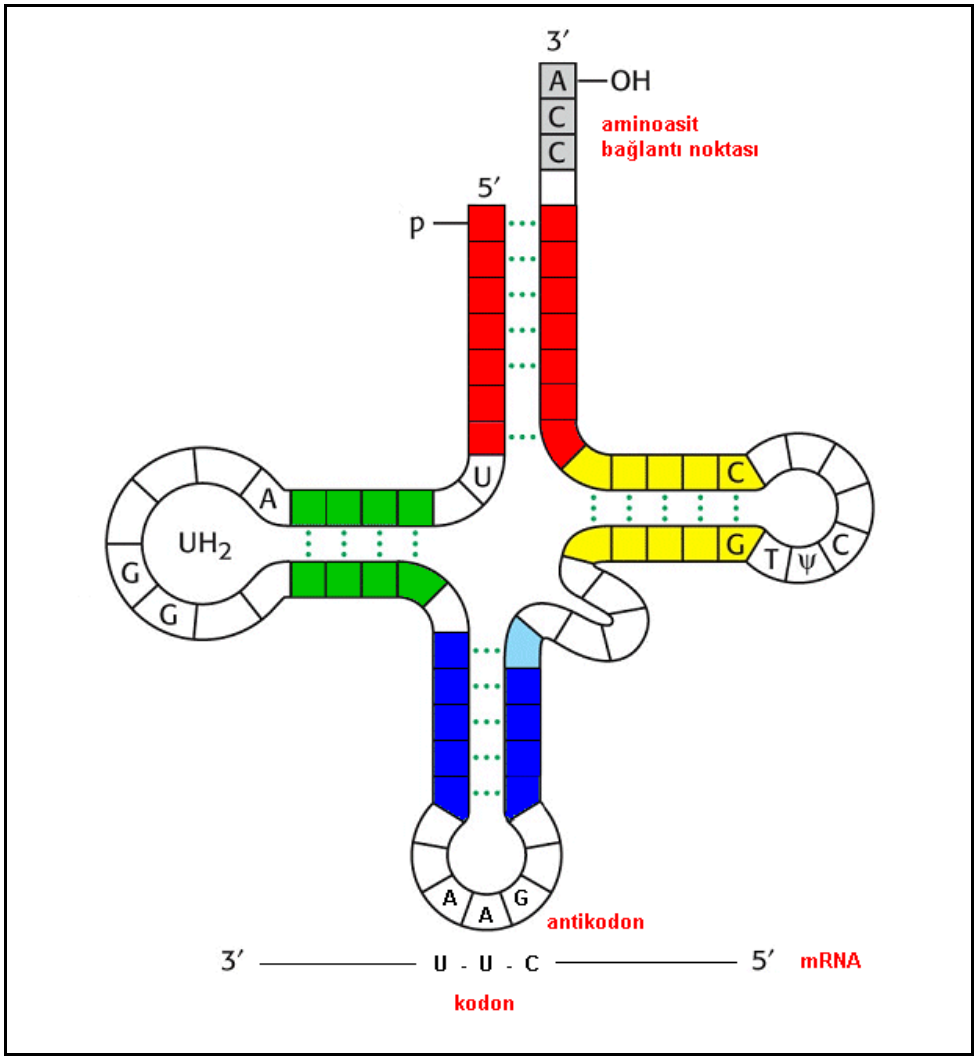
Genetik bilgi akışında sıra protein sentezine geldiğinde mRNA'dan başka tRNA da devreye girerek ribozomlarda protein sentezi gerçekleşir. mRNA da yer alan kodonların taşıdığı genetik mesaj ribozomlarda adım adım deşifre edilerek uygun amino asitler tRNA vasıtasıyla ribozoma getirilir. Hücre sitoplazmasında 20 çeşit aminoasil-tRNA ların ribozomda bağlanabilecekleri çeşitli bölgeler bulunur ve amino asitlerini bırakan tRNA'lar ribozomlardan ayrılırken polipeptid zinciri de sentezlenmiş olurlar. tRNA'lar üzerinde yer alan nükleotitlere antikodon adı verilir. Örneğin, UUU şeklinde olan bir mRNA zincirine uyan tRNA antikodonunun nükleotid sırası AAA şeklindedir. UUU şeklinde bir kodon da fenilalanin adlı aminoasitin şifresidir.

2.1.1 tRNA Yapısı ve İşlevi

TRNA, translasyon sırasında büyüyen polipeptid zincirine özel amino asitlerin eklenmesini sağlayan 74-93 nükleotid uzunluğunda küçük bir RNA zinciridir. Yapısında amino asit bağlanması için bir bölgesi ve mRNA üzerindeki kodon alanına karşılık gelen antikodon alanı vardır. Her tRNA molekülü sadece bir amino aside bağlanabilir. Fakat genetik kodun aynı amino asidi belirten birden çok kodon içermesi durumunda farklı antikodonları oluşturan birçok tRNA tipi aynı amino asidi taşıyabilir.

Farklı tRNA bölgeleri, hidrojen bağlarıyla birbirlerine bağlanmış haldedirler. Şekil 2-2'de görülen tRNA'nın 3' ucu CCA nükleotid dizisine sahiptir ve burası amino asitlerin bağlandığı bölgedir. Antikodonlar 3'->5' yönünde, mRNA'da kodonlar 5'->3' yönünde okunur. Örneğin, antikodon baz sırası 3'-GAA-5' ise, mRNA'daki kodon 5'-CUU-3' biçimindedir. mRNA'daki

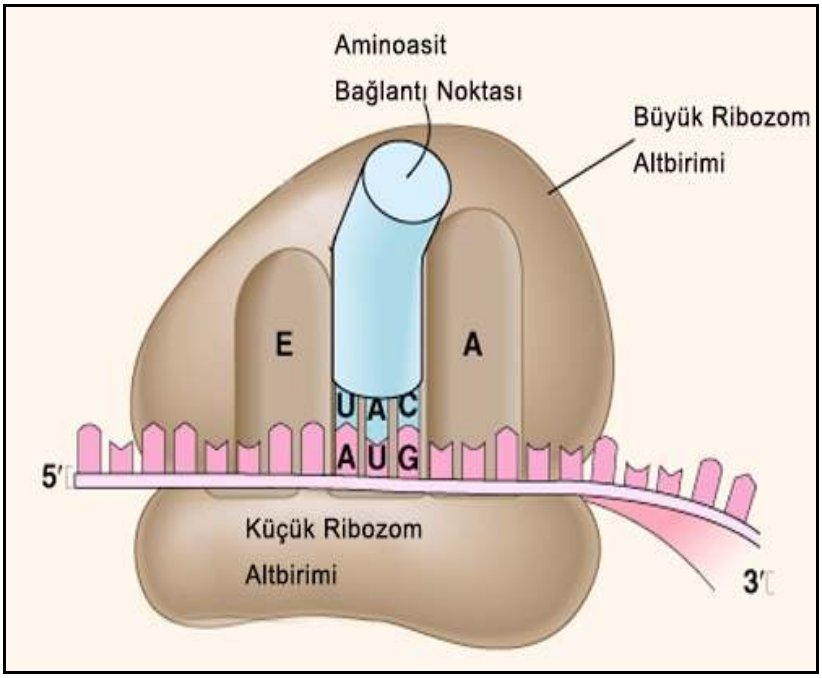
her bir amino asit kodonuna özgü bir tRNA olsaydı, 61 çeşit tRNA olması gerekirdi. Oysa tRNA çeşidi yaklaşık 45'tir. Bunun sebebi olarak, aynı antikodon bölgesine sahip olarak hazırlanan tRNA'ların, verilen amino asitlere uyumlu olarak birden çok kodonu tanıma yeteneğinde olduğu gösterilmiştir. Kodonların 3. pozisyonundaki baz ile onun antikodonundaki eşi olan 1. baz arasında standart olmayan bir baz eşleşmesi veya "serbestlik" özelliği nedeniyle bir tRNA çok sayıda kodonu tanıyabilir. Bu konuda en değişken tRNA, oynak pozisyonunda inosin (I) bulunduran tRNA'lardır. İnosin, 2. karbon atomunda amino grubu taşımayan bir guanin analoğudur. tRNA antikodonunun oynak pozisyonundaki inosin ile adenin, sitozin veya urasil ile eşleşebilir. Örneğin, tRNA antikodonu CCI olan bir tRNA, GGU, GGC ve GGA şeklindeki mRNA kodonlarına uyup, glisin amino asidini büyümekte olan protein zincirine katabilir.



Şekil 2-2 tRNA'nın yapısı

2.1.2 Ribozomlar

Ribozom protein sentezinin yapıldığı, mRNA ile tRNA'lar arasındaki bağlantının kurulduğu organeldir. Büyük ve küçük alt birim olmak üzere iki kısımdan oluşur, bunlar protein sentezi sırasında birleşirler. Ribozom, protein ve ribozomal RNA'lardan (rRNA) meydana gelmiştir. Ökaryotlarda alt birimler çekirdekçikte sentezlenir. Her bir ribozomda üç bağlanma bölgesi vardır. Polipeptide eklenmek için bekleyen aminoasil-tRNA, A yüzeyinde beklerken, sentezlenen polipeptid P yüzeyinde durur. Yükünü boşaltan tRNA ise ribozomdan çıkmak için E yüzeyine geçer. Bu işlemlerin olabilmesi için mRNA kodonları ile tRNA antikodonları arasındaki eşleşmelerin uygun olarak gerçekleşmesi gerekir. Prokaryot ve ökaryot ribozomları arasında benzerliklerle birlikte bazı farklılıklar da vardır. Bakterilere karşı kullanılan antibiyotiklerin bazıları spesifik olarak prokaryot ribozomlarına etki ederek protein sentezini, ve dolayısıyla bakterinin büyümesini durdururlar.



Şekil 2-3 Ribozomun yapısı

2.1.3 Transkripsiyon

Transkripsiyon için DNA çift sarmalının sadece bir iplikçiği gereklidir. Bu ipliğe "kalıp iplikçik" denir. Transkripsiyonun başlangıç noktasını tayin eden RNA polimeraz enzimi DNA üzerinde belirli bir bölgeye bağlanır. Bu bağlanma bölgesine promotor denir. RNA polimeraz promotora bağlandığında, DNA iplikçikleri açılmaya başlar. İkinci aşama uzamadır. RNA polimeraz, kodlamayan kalıp iplikçik üzerinde dolaşırken bir ribonükleotid polimeri sentezler.

RNA polimeraz kodlayıcı iplikçiği kullanmaz çünkü herhangi bir ipliğin kopyası, kopyalanan ipliğin tümleyici baz dizisini üretir. Polimeraz sonlanma aşamasına geldiğinde, RNA polimeraz, DNA ve yeni sentezlenmiş RNA birbirlerinden ayrılırlar. Prokaryotlardaki süreçten farklı olarak ökaryotlarda yeni sentezlenen mRNA'nın sitoplazma ve endoplazmik retikulum dahil birçok hücre bölgesine ulaşması için değişikliğe uğraması gerekmektedir. Yıkılmasını önlemek için mRNA'ya 5' başlığı eklenir. Kalıp olmak ve daha sonra işlenmesini sağlamak için 3' ucuna bir poli-A kuyruğu eklenir. Ökaryotlardaki hayati önem taşıyan uçbirleştirme olayı bu aşamada gerçekleşmektedir.

2.1.4 Translasyon

Protein yapımı (translasyon) üç aşamaya ayrılabilir: başlama, uzama ve sonlanma. Translasyon için mRNA, tRNA ve ribozomların yanı sıra bazı protein faktörleri de gereklidir. Enerji ise guanozin trifosfat'tan (GTP) sağlanır.

DNA'yı kaynak olarak kullanan RNA polimeraz enzimi tarafından üretilen mRNA molekülü, IF proteinlerinin yardımıyla önce ribozomun küçük altbirimine bağlanır. Daha sonra mRNA 5' ucundan okunmaya başlar. AUG kodonu protein sentezini başlatıcı kodondur. Bu kodona Met-tRNAi (bakterilerde fMet-tRNAf) molekülü bağlanır. Daha sonra ribozomun büyük alt birimi ile küçük alt birimi birleşir ve protein sentezi ilerler. Gerekli olan enerji GTP'den sağlanır. Başlatıcı kodona uyan tRNA, ribozomun P bölgesine yerleşerek A bölgesine kodona uygun yeni bir aminoasil-tRNA gelmesi beklenir.

Ribozomun A yüzeyine uygun antikodona sahip tRNA gelir ve hidrojen bağlarıyla kodona bağlanır. Bu sırada 2 molekül GTP harcanır. İkinci basamakta P yüzeyde bulunan polipeptid, A yüzeyine gelen amino asit ile birleşecek biçimde ortama aktarılır. Ribozom, mRNA üzerinde 3' yönüne doğru hareket ederek A yüzeyinde bulunan tRNA ile birlikte polipeptidi P yüzeyine aktarır. P yüzeyinde bulunan tRNA ise E yüzeyine geçerek ribozomdan uzaklaştırılır. Enerji GTP'den sağlanır. Ribozom, mRNA üzerinde 5'->3' yönünde hareket eder. Okuma ise kodon seviyesinde gerçekleşir.

Uzama, mRNA üzerinde durma kodonlarına kadar devam eder. A yüzeyine serbest bırakıcı faktörler geldiğinde okuma sonlanır. Bu faktörlerin A yüzeyine gelebilmesi için mRNA'daki kodonun UAG, UAA veya UGA şeklinde olması gerekir. Hidroliz enzimleri yardımıyla P yüzeyinde bulunan polipeptit serbest bırakılır. Böylece protein sentezi sonlanmış olur.

2.2 mikroRNA (miRNA)

Genetikte, mikroRNA (miRNA) yaklaşık 21-23 nükleotit uzunluğunda tek iplikçikli RNA

molekülü türüdür ve gen ekspresyonunda rol oynar. miRNA'lar kodlamayan RNA'lardandır, yani DNA'dan transkripsiyonu yapılan ama proteine çevirisi yapılmayan genler tarafından kodlanırlar. Pri-miRNA olarak adlandırılan primer transkriptler işlenerek, önce pre-miRNA adlı kısa sap-ilmik yapılarına, sonra da fonksiyonel miRNA'ya dönüşürler. Olgun miRNA moleküller bir veya daha çok mesajcı RNA (mRNA) ile kısmi tamamlayıcıdır ve başlıca işlevleri gen ifadesini aşağı ayarlamaktır. 1993'te Lee ve çalışma arkadaşları tarafından Victor Ambros laboratuvarında keşfedilmişlerdir, ancak mikroRNA terimi ilk 2001'de kullanıma girmiştir.

2.2.1 miRNA Oluşumu

miRNA'yı kodlayan genler işlenmiş olgun miRNA molekülünden çok daha uzundur. miRNA'lar önce birincil (primer) transkript, veya pri-miRNA olarak yazılırlar, bu transkriptlerin birer başlığı ve poli-A kuyruğu vardır. Bunlar işlem görüp hücre çekirdeğinde pre-miRNA olarak bilinen kısa, 70 nükleotit uzunlukta, sap-ilmik şekilli, öncül (prekürsör) yapılara dönüşür. Bu işlenme hayvanlarda "mikroişlemci kompleks" (İng. microprocessor complex) adlı bir protein kompleksi tarafından gerçekleştirilir. Mikroişlemci kompleks'te Drosha adlı bir nükleaz, ve Pasha adlı çift iplikçikli RNA bağlayıcı protein bulunur.

Pre-miRNA'lar sonra sitoplazmada Dicer adlı endonükleaz ile etkileşerek olgun miRNA'ya dönüşürler. Dicer aynı zamanda RNA-indüklenmiş susturma kompleksi (İng. RNA-induced silencing complex; RISC) oluşumunun başlatır. Bu kompleks miRNA ifadesi ve RNA interferanstan kaynaklanan gen susturmasından sorumludur.

Bitkilerde miRNA oluşumunun yolu, Drosha'nın homologlarının olmamasından dolayı hayvanlardakinden biraz farklıdır; onun yerine Dicer homologları tek başlarına birkaç işlem aşamasını yürütürler. Ayrıca intronik sap-ilmiklerden meydana gelen miRNA'ların oluşumunda da Drosha değil, Dicer görev alır. DNA'nın anlamlı veya ters anlamlı iplikçiği de miRNA'nın oluşumunda kalıp işlevi görebilir.

Pre-miRNA'nın verimli bir şekilde işlenmesi için, sap-ilmik yapısının hem 5', hem de 3' ucunda tek iplikçikli RNA uzantıların olması gerekir. Bu tek iplikçikli RNA motiflerin bileşimlerinden çok uzunlukları son derece önemlidir, işlenmenin gerçekleşebilmesi için. İnsan ve sinek pri-miRNA'larının bir biyoenformatik analizi, çok benzer yapısal bölgelerin varlığını göstermiş, bunlar, bazal kısımlar, aşağı saplar, yukarı saplar ve uç ilmikler olarak adlandırılmıştır; evrimsel olarak korunmuş bu yapılara dayanarak pri-miRNA'nın termodinamik profilleri tanımlanmıştır. Drosha kompleksi, RNA molekülünü uç ilmikten yaklaşık 22 nükleotit uzaktan keser. Çoğu pre-miRNA'da tepesinde bir ilmik olan mükemmel

çift sarmallı bir yapı yoktur. Bu seçiciliğin birkaç olası açıklaması vardır. Bir olasılık, 21 baz çiftinden uzun çift sarmallı RNA'nın enterferon tepkisi ve hücrenin anti-viral mekanizmasını harekete geçirmesidir. Bir diğer makul açıklama pre-miRNA'nın termodinamik profilinin hangi iplikçiğin Dicer kompleksine dahil olacağını belirlemesidir.

Dicer pre-miRNA sap-ilmiğini kestikten sonra iki tamamlayıcı kısa RNA molekülü meydana gelir, ama bunlardan sadece biri RISC kompleksine dahil olur. RISC kompleksinin içinde yer alan bir RNAz olan argonaute'un etkisiyle bu ikisinden 5' ucu daha kararlı olanı seçilip komplekse dahil olur. Bu iplikçik kılavuz iplikçik (İng. guide strand) olarak adlandırılır. Öbür iplikçik, anti-kılavuz veya yolcu iplikçik olarak adlandırılır, RISC kompleksinin substratı olarak sindirilir. Aktif RISC kompleksine entegre olduktan sonra miRNA'lar kendi tamamlayıcı mRNA molekülleri ile baz eşleşmesi yapar ve argonaute proteinleri tarafından mRNA'nın yıkımına neden olurlar.

2.2.2 miRNA Hücresel İşlevi

miRNA'nın işlevinin gen düzenlemesi olduğu anlaşılmaktadır. Bu amaçla bir miRNA bir veya daha çok mRNA'yı tamamlayıcıdır (komplemanterdir). Hayvan miRNA'ları genelde 3' UTR bölgesine tamamlayıcıdırlar, bitki miRNA'ları ise mRNA'ların protein kodlayıcı kısımlarına tamamlayıcıdır. miRNA'nın mRNA ile eşleşmesi bazen protein çevirisini engeller ve bazen de mRNA'nın kesilmesini (RNA interferansa benzer bir süreçle) kolaylaştırır. Çoğu hayvan miRNA'sı protein çevirisini engeller, çoğu bitki miRNA'sı da mRNA'yı keserek çalışır. miRNA'lar ayrıca hedef mRNA'ya karşılık gelen genomik bölgelerde DNA metilasyonuna neden olabilirler. miRNA'lar kendilerini tamamlayan bir grup proteinle (mikroribonükleoproteinler = miRNP) birlikte işlev görürler.

miRNA'nın etkileri ilk 1993'te Victor Ambros ve çalışma arkadaşları tarafından C. elegans solucanında keşfedildi. miRNA'ların varlığı çeşitli bitki ve hayvanlarda teyid edilmiştir. Ökaryotik miRNA genlerinin benzerleri bakterilerde de bulunmuştur, bunlar mRNA ile eşleşerek mRNA çokluğunu ve çevirisini kontrol etmektedirler, ancak bu süreçte Dicer enziminin bir benzeri yer almadığı için bu RNA'lar genel olarak miRNA olarak sayılmamaktadırlar.

Bitkilerde kısa müdahaleci RNA'lar (İng. short-interfering RNA; siRNA kısaltmasıyla anılır) viral RNA'nın transkripsiyonunu engellemeye yarar. siRNA çift iplikçikli olmasına rağmen, etki mekanizması miRNA'nınkine yakından benzer, saç firkete yapıları göz önüne alınırsa bu benzerlik daha da çarpıcıdır. siRNA'lar, miRNA'lar gibi, gen denetimine yararlar.

3. MAKİNE ÖĞRENMESİ

Bilgisayarların veritabanları veya çeşitli algılayıcı ağlardan elde ettikleri verileri kullanarak öğrenmesini sağlayan algoritmaların tasarımı ve geliştirilmesi makine öğrenmesi olarak özetlenebilir. Makine öğrenmesi araştırmalarındaki en can alıcı nokta geliştirilen algoritmanın çok karmaşık yapıdaki örüleri kendi kendine algılayıp bunlarda elde ettiği veriler ışığında akıllı kararlar verebiliyor olmasıdır. Dolayısıyla makine öğrenmesi ; istatistik, olasılık teorisi, veri madenciliği, örüntü tanıma, yapay zeka gibi disiplenlerle yakından ilişki içerisindedir. Bu sayede günümüze kadar ortaya konmuş birçok makine öğrenmesi yöntemi bulunmaktadır.

3.1 Öğrenme Yöntemleri

Makine öğrenmesi yöntemleri aşağıdaki başlıklarda incelenmiştir.

3.1.1 Eğitmenli (Supervised) Öğrenme

Eğitmenli öğrenme, öğrenme verilerinden bir fonksiyon oluşturma esasına dayanan makine öğrenmesi yöntemidir. Öğrenme verileri, giriş değerlerini (genelde vektörler) ve bu giriş değerlerine karşılık gelen çıkış değerlerini içermektedir. Eğitmenli öğrenme yönetiminin görevi öğrenme verilerini kullanarak kendi oluşturacağı fonksiyonun katsayılarını fonksiyonu her durumda doğru sonuç üretecek şekilde tahmin edebilmesidir. Bunu yapabilmek için öğrenici elindeki giriş verilerini kullanarak genellemeler yapmak durumundadır.

Eğitmenli öğrenmede izlenmesi gereken adımlar şu şekilde sıralanabilir :

1. **Öğrenme verilerinin tipinin belirlenmesi :** Bu ilk adımda verilerin nasıl kullanılacağına karar verilir. Bir RNA dizisi için bu bir nükleotid, bir kodon veya bir geni ifade eden dizilimin tamamı olabilir.
2. **Öğrenme verilerinin elde edilmesi :** Öğrenme verileri gerçek hayattaki örnekleri birebir yansıtacak nitelikte seçilmelidir. Makine öğrenmesi açısından bunun önemi çok büyüktür. Eğer seçilen veriler gerçek hayattaki deneyimleri yansıtmıyorsa öğrenme fonksiyonu doğru sonuçlar üretmeyecektir.
3. **Öğrenme fonksiyonu giriş verisi özelliklerinin belirlenmesi :** Öğrenme fonksiyonunda öğrenme veri kümesinde yer alan her bir nesneye karşılık gelen ham veri yerine bu verideki ayırt edici özellikler kullanılır. Örneğin, bir mRNA::miRNA eşleşmesinde A-U eşleşmelerinin toplam eşleşmelere oranı ayırt edici bir özellik olarak kullanılabilir. Özellik sayısının yüksek olması öğrenmeyi karmaşıklaştırabilir. Öte yandan özellik sayısının az olması da doğru öğrenmeyi

engelleyebilir. Bu yüzden özellik sayısının seçiminde dikkatli olunmalıdır.

4. **Öğrenme algoritmasının belirlenmesi :** Bilinen öğrenme yöntemlerinden biri seçilir. Örneğin yapay sinir ağları veya karar ağaçları.
5. **Öğrenmenin gerçekleştirilmesi :** Belirlenen öğrenme algoritması elde edilen öğrenme verileri ile çalıştırılır. Öğrenme fonksiyonu katsayıları çeşitli doğrulama yöntemleri kullanılarak en uygun hale getirilebilir. Katsayıların ayarlanmasından sonra oluşan fonksiyon sınama verileri ile sınanır. Elde edilen sonuçlarla algoritmanın başarımı ve doğruluğu hesaplanabilir.

3.1.2 Yarı-Eğitmenli (Semi-supervised) Öğrenme

Yarı eğitmenli öğrenme algoritmaları genelde az miktarda etiketlenmiş ve çok miktarda etiketlenmemiş öğrenme verisi kullanır. Birçok araştırmacı etiketlenmemiş verinin makine öğrenmesine olumlu yönde etki ettiğini tespit etmiştir. Etiketlenmemiş veri beklenen çıktı değerinin olmadığı veriyi temsil eder. Bu yüzden etiketlenmemiş veri ile eğitilen öğreniciler insan müdahelesi olmadan öğrenmeyi gerçekleştiremez. Genelde konusunun uzmanı kişiler tarafından elle müdahele edilerek öğreniciler eğitilir.

3.1.3 Eğitmensiz (Unsupervised) Öğrenme

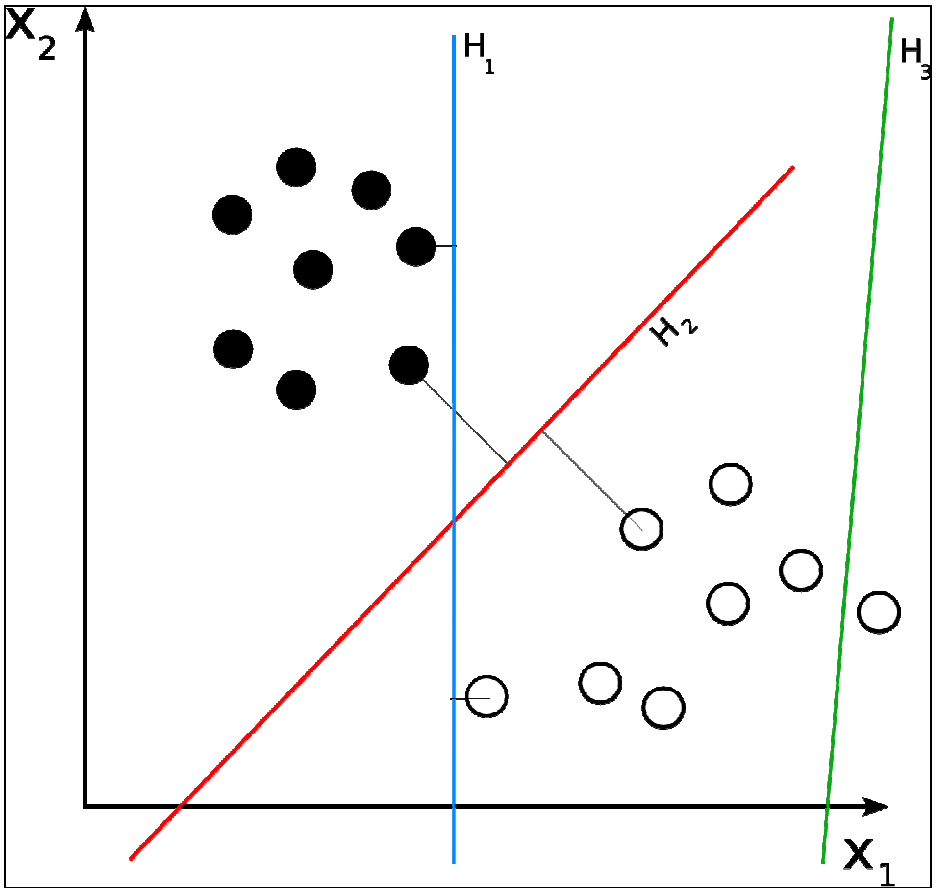
Makine öğrenmesinde tamamen etiketlenmiş veri kullanan yöntemdir. Eğitmensiz öğrenmenin bir şekli kümelemedir. Kümeleme algoritmaları belirli bir eşik değeri ışığında belirlenen yöntemleri eldeki verilere uygulayarak öğrenme verisini sınıflara böler. Yeni bir veri girildiğinde bu verinin hangi sınıfa dahil olacağı hesaplanır.

3.2 Destek Vektör Makineleri

Destek Vektör Makineleri (DVM) sınıflandırma ve regresyon için kullanılan makine öğrenmesi yöntemleri bütünüdür. Bu konuda ilk çalışma 1992 yılında Vladimir N. Vapnik ve arkadaşları tarafından yapılmış ve o günden bu yana birçok araştırmaya konu olmuştur.

N-boyutlu uzayda giriş verileri iki vektör kümesi olarak alındığında DVM bu iki vektör kümesinin aralarındaki boşluk en yüksek seviyede olacak şekilde bir hiperdüzlem tanımlar. En uygun hiperdüzlem her iki veri kümesinde kendine en yakın iki elemanın hiperdüzleme olan uzaklıkları toplamının en yüksek olduğu hiperdüzlemdir. Şekil 3-1'de görülen H_1 , H_2 ve H_3 hiperdüzlemleri incelenecek olursa H_1 ve H_2 hiperdüzlemlerinin iki veri kümesini ayırdığı, H_3'ün ise bu iki veri kümesini ayıramadığı görülmektedir. H_1 için iki küme

arasındaki uzaklık kümelere ait H_1'e en yakın elemanların H_1'e olan uzaklıkları toplanarak hesaplanır. Aynı uzaklık H_2 için de hesaplanacak olursa H_2'nin iki veri kümesi için en uygun ayırıcı hiperdüzlem olduğu görülür.



Şekil 3-1 İki veri kümesini ayıran hiperdüzlemler [2]

DVM üzerinde yapılan ilk çalışmalarda hedef ikili sınıflandırma yapmaktı. Ancak daha sonra yeni sürümleri ortaya çıkarılarak çoklu sınıflandırma yapabilen DVM'ler oluşturuldu.

DVM ikili sınıflandırma için kullanıldığında sonucu {+1,-1} kümesi olan bir karar fonksiyonu sağlar:

$$f(x,\alpha,b) = \{\pm 1\} = \mathrm{sgn}(\sum_{i=1}^{l} \alpha_i y_i k(x_i, x) + b) \qquad (3.1)$$

Fonksiyondaki α değeri (3.2) optimizasyon problemi çözülerek bulunur.(Vapnik vd.)

$$\text{maximize } W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j k(x_i, x_j) \tag{3.2}$$

$$0 \le \alpha_i \le C \tag{3.3}$$

$$\sum_{i=1}^{l} y_i \alpha_i = 0 \tag{3.4}$$

(3.2) fonksiyonun çıktısı ±1 olabilir. Buna göre giriş verisi "+" veya "–" sınıfına ait olur. DVM giriş verisine ait özelliklerin kombinasyonundan oluşan bir özellik uzayında yer alan noktalar ile oluşturulur. Bu noktalara genellikle çekirdek veya içsel ürün denir. (3.3) ve (3.4) Karush-Kuhn-Tucker (KKT) kısıtları olarak nitelendirilir. KKT kısıtları dikkate alınarak DVM in çözülebilmesi için $\alpha_i > 0$ olmalıdır. Her bir öğrenme verisi, $x_{i \in 1..l}$ - veya + sınıfına aittir. Bu değer y_i 'de tutulur.

DVM'nin hesaplama yükü destek vektörlerinin sayısına bağlıdır. Veri miktarı arttıkça hesaplama zamanı da artar. DVM parametrelerini en uygun seviyeye çekebilmek için çeşitli doğrulama yöntemleri kullanılır. Bunlar içinde en çok LOOCV (Leave-one-out Cross Validation) kullanılır. LOOCV verilen n adet vektörden birini giriş kümesinin dışında tutup arta kalan n-1 vektör ile DVM'i eğitir. Dışarıda kalan vektör ise DVM'i sınamak için kullanılır. Bu işlem giriş vektör kümesindeki her eleman için tekrarlanır.

3.2.1 Öğrenme

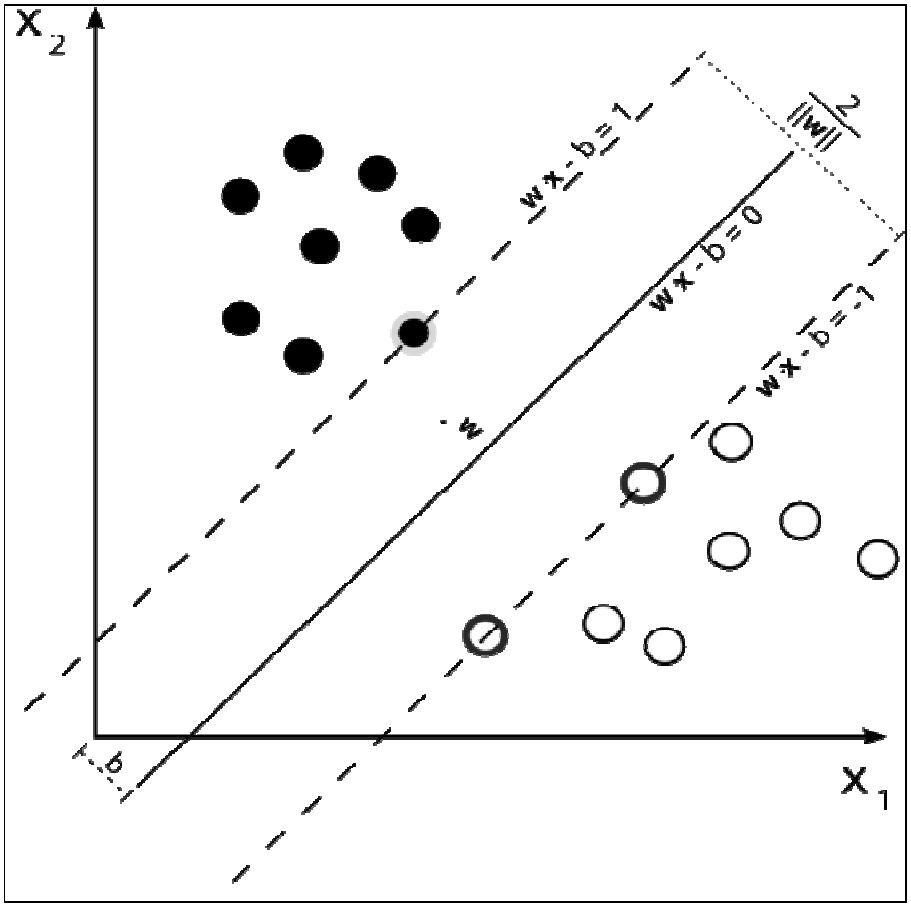
DVM öğrenme süreci bir çeşit sınıflandırma yöntemidir. Sınıflandırma, giriş verisinden elde edilen özellik kümesinin oluşturduğu özellik uzayında gerçekleştirilir. Bu özellik uzayında giriş verisine denk düşen her nokta bir destek vektörü olarak adlandırılır.

DVM her ne kadar çoklu sınıflandırmada kullanılabilse de genellikle ikili sınıflandırmada kullanılır. İkili sınıflandırma bir destek vektörünün {± 1} ile ifade edilen kümelerden hangisine ait olduğunu bulmak amacıyla yapılır. Öğrenme aşamasında "+" veya "-" kümeye ait olduğu bilinen vektörler giriş verisi olarak verilir. DVM sınıflandırma yöntemi olarak kullandığı fonksiyon ile verilen vektörlere ait küme değerlerini hesaplar. Bu değerler kullanılan fonksiyonun türüne göre değişir.

3.2.1.1 Doğrusal Çekirdek Fonksiyonu

Doğrusal çekirdek öğrenme verisi olarak verilen kümeyi özellik kümesinin boyutuna bağlı olarak doğru, düzlem veya hiperdüzlem olarak ayıran bir sınıflandırma fonksiyonu sağlar.

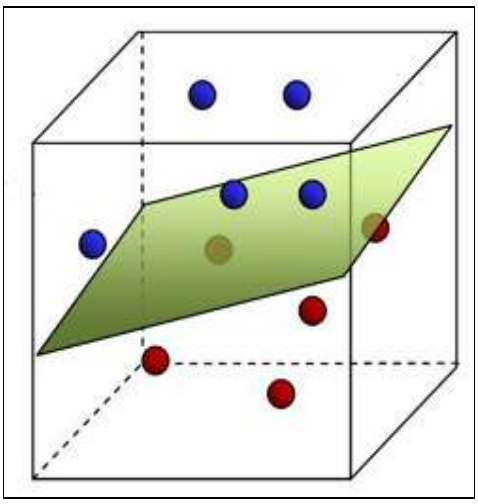
Şekil 3-2'de görülen veri kümesi iki özellik ile ifade edilen vektörlerden oluşmaktadır. Bu vektörleri ayıran sınıflandırma fonskiyonu da doğrusal çekirdek kullanmaktadır ve sınıfları ayıran bir doğru parçasıdır.



Şekil 3-2 İki boyutlu veri kümesini doğrusal çekirdek fonksiyonu [4]

Şekil 3-3'de görülen veri kümesi üç farklı özellikle ifade edilen vektörlerden oluşmaktadır. Bu vektörlerin arasında bir sınır çizilebilmesi için bir düzleme gerek duyulmaktadır. Bu düzlem görüldüğü gibi yine doğrusal çekirdek fonksiyonu tarafından sağlanabilmektedir.

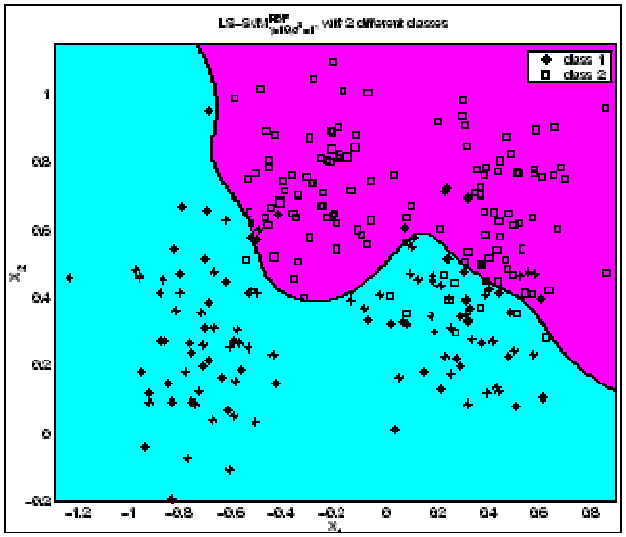
Verilen örneklerin ötesine gidilecek olursa, doğrusal bir çekirdek fonksiyonu teorik olarak sonsuz sayıda özellik içeren bir vektör kümesini bile ayırmak için yeterli olacaktır. Ancak doğrusal çekirdek fonksiyonları her zaman yeterli seviyede sınıf ayrımı yapamamaktadır. Çünkü "+" ve "-" kümeler arasındaki ayrım her zaman doğrusal olmayabilir. Bu yüzden yeni çekirdek fonksiyonlarının tanımlanmasına gerek duyulmuştur.



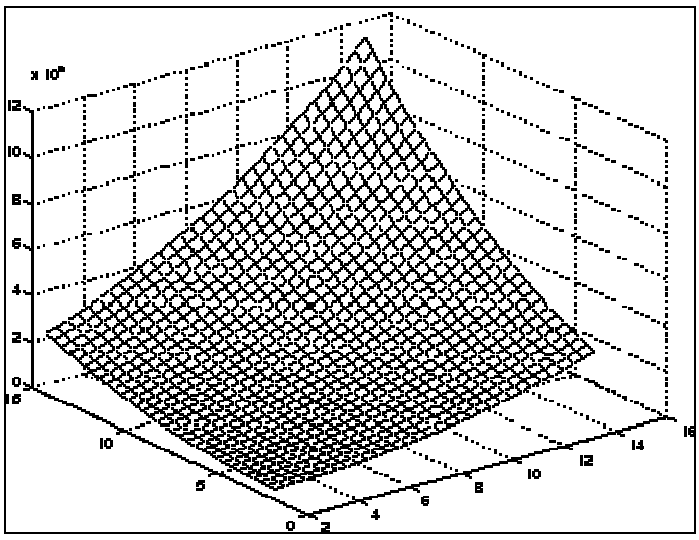
Şekil 3-3 Üç boyutlu veri kümesini ayıran doğrusal çekirdek fonksiyonu. [5]

3.2.1.2 Polinom Çekirdek Fonksiyonu

Doğrusal bir ayıraç fonksiyonun yeterli olmadığı veri kümeleri polinomlarla ayrılabilir. Şekil 3-4'de görülen iki farklı özellikle ifade edilmiş veri kümesini ayıran bir polinomdur. DVM öğrenme sırasında bu polinoma ait denklemin katsayılarını bulmaya çalışır. Şekil 3-5'da üç boyutlu bir veri kümesini ayıran polinom görülmektedir.



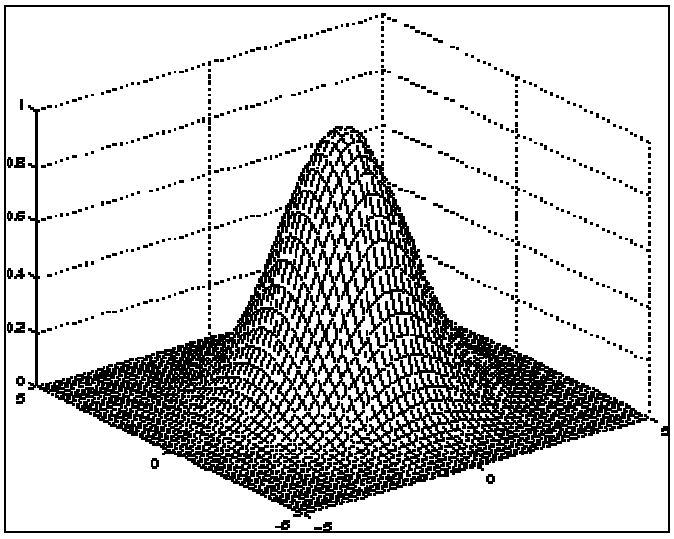
Şekil 3-4 İki boyutlu veri kümesini ayıran polinom çekirdek fonksiyonu [6]



Şekil 3-5 Üç boyutlu veri kümesini ayıran polinom çekirdek fonksiyonu [7]

3.2.1.3 RBF Çekirdek

RBF (Radial Basis Function) n-boyutlu uzayda bir noktadan belirli mesafedeki noktaları tanımlayan her fonksiyona verilen isimdir. Şekil 3-6'de görülen basit, temel bir RBF görülmektedir. RBF çekirdek genelde birden fazla fonksiyonun toplamından oluşur.



Şekil 3-6 RBF çekirdek fonksiyonu [8]

$$K(x_i, x_j) = \exp(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma^2}) \quad (3.4)$$

RBF'nin matematiksel ifade şekli (3.4) ile verilmiştir. Fonksiyondaki σ parametresi DVM'in özgüllük ve doğruluk değerlerini etkiler. Büyük σ değerleri daha pürüzsüz bir karar yüzeyi ve daha düzgün bir karar çizgisi sağlar. Çünkü büyük σ değerleri destek vektörlerine daha büyük bir veri kümesi üzerinde etki etme olanağı verecektir.

3.2.2 Sınama

DVM öğrenme süreci tamamlandıktan sonra, öğrenilen değerlerin doğruluğunun sınanması ve onaylanması gerekmektedir. Bu süreçte değişik yöntemler kullanılmaktadır. Bu yöntemler aşağıdaki başlıklarda incelenmiştir.

3.2.2.1 HOV (Hold-Out Validation) Yöntemi

HOV en çok kullanılan sınama yöntemlerinden biridir. Örnek olarak verilen veri kümesini iki parçaya bölüp parçalardan birini öğrenme amaçlı diğerini ise sınama amaçlı kullanma esasına dayanır. Genelde örnek kümesinin 1/3'ü sınama için kalan kısmı da öğrenme için kullanılır. Verilen örnek kümesi iki ayrı parçaya bölündüğü için öğrenme ve sınama veri kümeleri ortak elemanlar içermez. Örnek kümesi bölünürken dikkat edilmesi gereken en önemli nokta bölünme sonrasında her iki veri kümesinin de ana kümeyi temsil edebilecek nitelikte elemanlar içeriyor olmasıdır. Bölünmenin ne kadar sağlıklı olduğu sınıflandırmanın da ne kadar doğru olacağını belirler.

Bu yöntem yapısı gereğince uygulamada ve hesaplamada basittir. Ancak küçük örnek veri kümeleri söz konusu olduğunda HOV güvenilir ve uygulanabilir bir yöntem değildir.

3.2.2.2 KFCV (K-fold Cross Validation) Yöntemi

KFCV yöntemi verilen örnek veri kümesinin eleman sayıları eşit olan k tane alt kümeye bölünerek öğrenme ve sınama süreçlerinin k kez tekrar edilmesi esasına dayanır. Bölünme sırasında elemanlar rastgele dağıtılır. Her yinelemede alt kümelerden biri sınama kalanları da öğrenme için kullanılır. Öğrenmenin doğruluk değeri her bir yinelemenin doğruluk değerlerinin ortalaması alınarak hesaplanır.

3.2.2.3 LOOCV (Leave-one-out Cross Validation) Yöntemi

Günümüzde en yaygın kullanılan sınama yöntemidir. Örnek kümesindeki n elemandan $n-1$ tanesi eğitim ve 1 tanesi sınama için kullanıldığında bu işlem kümedeki her eleman için gerçekleştirilirse LOOCV yöntemi uygulanmış demektir. Bu yöntem kullanıldığında örnek verileri en iyi şekilde kullanılmış olacaktır. KFCV yöntemi düşünüldüğünde $k=n$ olduğu görülür. Bu yüzden LOOCV, KFCV'nin özel durumdaki bir türevidir. Hesaplama maaliyeti olarak en pahalı sınama yöntemidir. Tüm örnek kümesi düşünülürse sınama sırasında n^2 işlem yapılır.

4. PMIRNA SİSTEMİ

Xue vd. öncül miRNA (pre-miRNA) sınıflandırmasında RNA dizilimine ait yapısal özelliklerini DVM'i eğitmek için kullanmışlardır. Bu çalışmada yenilik olarak ortaya atılan ve yerel sürekli dizilim olarak nitelendirilen yapısal özellik, ard arda gelen üç nükleotidin (kodon) karşılarındaki nükleotidlerle nasıl etkileştiğini ifade eden gösterim biçimidir. Buna göre eğer eşleşen bir nükleotid çifti varsa bunlar ilk nükletoid için "(" ve karşılık gelen nükleotid için ")" şeklinde, eğer eşleşmeyen bir nükleotid çifti varsa "." şeklinde ifade edilir. Bu durumda her bir kodon için 8 (2^3) farklı eşleşme olasılığı bulunmaktadır. Bunlar "(((", "((.", "(..", "(.(", ".((", ".(.", "..(" ve "..." dir. Öne sürülen bu yeni yöntemin doğruluk başarısı %90 olarak verilmiştir. (Chenghai Xue vd., 2005)

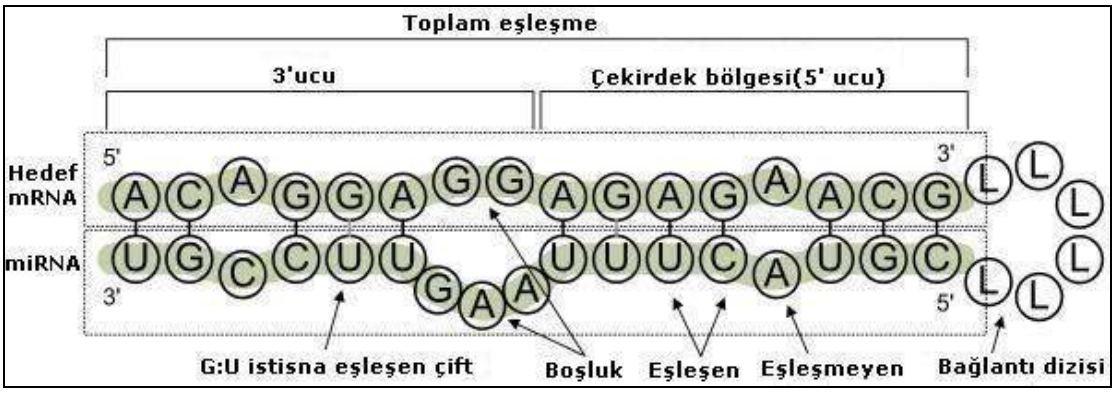
Kim vd. geliştirdikleri miTarget isimli yazılım aracı ile miRNA veri kümesine ait yapısal, termodinamik ve pozisyon tabanlı özellikleri kullanarak miRNA hedef genlerini tahmin etmeye yönelik bir DVM oluşturmuşlardır. Bu çalışmada nükleotid eşleşmeleri tek tek alınmış ancak eşleşmelerin nasıl olduğu farklı değerlendirilmiştir. Önceleri eşleşmelerin G-C ve A-U çiftleri şeklinde olabileceği düşünülürken G-U eşleşmelerinin de olabileceği görülmüştür. Fakat G-U eşleşmesi G-C ve A-U eşleşmelerine göre biyolojik olarak daha zayıf bir eşleşmedir. Zira G (Guanine) – C (Cytosine) eşleşmesi 3 hidrojen bağı, A (Adenine) – U (Urasile) eşleşmesi 2 hidrojen bağı ile oluşurken G – U eşleşmesi yalnızda 1 hidrojen bağından meydana gelir. Bu eşleşmelerin dışında istisnai eşleşmeler de meydana gelebilmektedir. Sonuç olarak yapısal özellikler eşleşme olan çiftler ve eşleşme olmayan (yanlış eşleşen) çiftlerden meydana gelen bir özellik kümesi oluşturur. Bu kümede G-C ve A-U eşleşen çiftler, diğer çiftler de eşleşmeyen çiftler olarak kabul edilir. Termodinamik özellikler ise verilen RNA dizilimlerinin minimum enerjisinin hesaplanması ile ortaya çıkmaktadır. Bu hesaplama Vienna RNA Package isimli bir paket program yardımı ile yapılmıştır. Pozisyon tabanlı özellikler de eşleşmelerin başladığı ve bittiği noktalar temel alınarak her bir pozisyon için gözlenen eşleşme türüne göre belirlenir. Eşleşme türleri yapısal özelliklerin çıkarılmasında kullanılan eşleşme türleri ile aynıdır. (Sung-Kyu Kim vd., 2006)

Liu vd. yaptıkları çalışmada farklı türlere ait miRNA:HedefGen etkileşimine ait veriler içeren TarBase isimli veritabanını kullanarak DVM için 200 çiftten oluşan bir öğrenme verisi oluşturmuşlardır. DVM eğitiminde kullanılan özellik kümesi (Sung-Kyu Kim vd., 2006) nin kullandıkları ile benzerlik göstermektedir. Her miRNA::HedefGen çifti için bölgesel ve pozisyon tabanlı özellikler belirlenmiştir. Bölgesel özellikler eşleşme olan ve olmayan çiftlerden çıkarılan özelliklerdir. Bir RNA dizisinde yer alabilecek yalnızca 4 çeşit nükleotid

olduğuna göre kaç farklı türde eşleşme olabileceği 4! = 4*3*2*1 = 16 şeklinde hesaplanabilir. Sonuç olarak (4.1) teki özellik kümesi ortaya çıkar.

$$F = \{"AA","AC","AG","AU","CA",...,"GG","GU","UA","UC","UG","UU"\} \quad (4.1)$$

Liu vd. oluşan yanlış eşleşmelerin de DVM'in doğru eğitilebilmesi için tek tek özellik olarak ele alınması gerektiğini belirtmiştir. Bölgesel özellikler eşleşme olan tüm bölge ve eşleşmenin merkezi denebilecek çekirdek bölge için ayrı ayrı hesaplanmıştır. Çekirdek bölge en fazla 8-10 nükleotid uzunluğunda olabilir. Buna göre tüm eşleşme 8-10 nükleotid içerecek şekilde bir kayan pencere algoritması ile baştan uca taranarak ard arda eşleşen en uzun nükleotid dizisi bulunur. Şekil 4-1'te bu dizi çekirdek bölgesi olarak belirtilmiştir. Pozisyon tabanlı özellik içinse eşleşmenin tümü göz önüne alınarak her bir pozisyon için oluşan eşleşmenin türü belirlenmiştir. (Hui Liu vd., 2008)



Şekil 4-1 miRNA::HedefGen Eşleşmesi [3]

Mitra ve Bandyopadhyay gelişmiş bir özellik kümesi kullanarak miRNA hedef gen tahminini iyileştirmeye yönelik bir çalışma gerçekleştirmişlerdir. Bu çalışmada DVM için negatif öğrenme verisi oluşturmada karşılaşılan sıkıntılara yer verilmiştir. Negatif verinin eldesi laboratuvar şartlarında neredeyse mümkün denmeyecek kadar zordur. Çünkü iki RNA dizisinin etkileştiğinin ispatı etkileşmediğinin ispatından çok daha kolaydır. Bu yüzden kesinlikle eşleşmediği bilinen miRNA:HedefGen çiftlerinin sayısı oldukça azdır. İşte bu sıkıntıyı gidermek amacıyla DVM'lerin eğitimlerini biraz olsun kolaylaştırabilmek için sanal negatif veriler oluşturulmaktadır. Farklı alanlarda yapılan incelemelerde DVM eğitirken verilen negatif veriler genelde bilgisayar tarafından oluşturulan rastgele değerlerdir. Ancak

genetik alanında rastgele değerler üretirken bazı kısıtlarla karşılaşılmaktadır. Rastgele üretilen değerler DVM'in doğruluk derecesinin düşmesine sebebiyet verebilmektedir. Bu yüzden üretilen negatif verilerin biyolojik olarak mümkün nitelikler taşıması gerekmektedir. Ramkrishna Mitra ve Sanghamitra Bandyopadhyay geliştirdikleri K-Mer yöntemi ile eşleşen miRNA::HedefGen çiftlerinden yola çıkarak çekirdek bölge ile bunun dışında kalan bölgedeki nükleotidleri çapraz olarak değiştirmişlerdir. Oluşan yeni çiftler negatif veri olarak DVM'e verilmiştir. (Ramkrishna Mitra ve Sanghamitra Bandyopadhyay, 2009)

Bu zamana kadar yapılan tüm çalışmalar daha iyi sonuçlar elde edebilmek için veri kümelerinin iyileştirilmesi yönünde olmuştur. Bunun yanında DVM üzerinde farklı sınama yöntemleri uygulanarak DVM'in parametrelerinin daha uygun hale getirilmesi üzerinde de çalışılmıştır. Kullanılan veri kümeleri genellikle tek tek olmak üzere elle seçilmiş dolayısıyla da az sayıda örnek içeren özel veri kümeleri olmaktan öte gidememiştir. Bunun sebebi üretilen verilerin bazı özellikler açısından yetersiz oluşudur. Ancak teknolojik gelişmelerin hızı göz önüne alınacak olursa yakın bir gelecekte bu alanda üretilen verilerin ciddi bir oranda artacağı, içerik olarak daha zengin ve hassas olacağı yadsınamaz bir gerçektir. Artan veri miktarı bu verilerin işlenmesini ve anlamlı bilgiye dönüştürülmesini de zorlaştıracaktır. Hesaplama karmaşıklığı yüksek olan DVM algoritmaları yüksek miktarda veri karşısında çaresiz yüksek işlem gücü gerektirecektir.

PMirna sistemi paralel hesaplama yöntemi kullanılarak geliştirilmiş DVM uygulamaları ile miRNA'ların hedef aldığı genleri tahmin etmeye yönelik bir çalışmadır. Bu çalışmanın hedefi artan veri miktarı ile birlikte uzun hesaplama süresi gerektiren DVM uygulamalarının paralel programlama sayesinde daha kısa sürelerde doğru sonuç vermesini sağlamaktır.

DVM uygulamalarını güçlü kılan doğru öğrenme verisi sağlandığında çok yüksek doğruluk ve hassasiyette sonuç üretebiliyor olmalarıdır. Doğruluk değerinin yüksek olması için DVM parametreleri değiştirildiğinde işlem karmaşıklığı artmaktadır. Bu noktada paralel çalışabilen DVM uygulamaları ön plana çıkmaktadır. PMirna sistemi, DVM için özellik çıkarmadan DVM testine kadar tüm süreçlerde paralel programlamadan mümkün olan en yüksek seviyede faydalanan bir sistemdir. PMirna 3 ana parçadan oluşur : Veri Toplama ve Özellik Çıkarma, Öğrenme ve Test, Tarama/Sorgulama.

4.1 Veri Toplama ve Özellik Çıkarma

miRNA üzerine çalışma yapan bir çok grup bu çalışmalarını Internet üzerinden yayınlamaktadır. Elde edilen veriler çok çeşitli formatlarda olmasına rağmen genel kabul

görmüş bazı formatlar da vardır. Bu tezde miRNA:HedefGen etkileşim bilgileri DIANA LAB [1] çalışma grubunun yayınladığı TarBase isimli veritabanı kullanılarak elde edilmiştir. Şekil 4-2'de görüldüğü gibi TarBase veritabanından "Homo sapiens" (insan) türüne ait miRNA:HedefGen etkileşim verileri ayıklanarak DVM özellik verilerinin hesaplanmasına uzanan bir süreçten geçilmektedir. Veritabanında miRNA'ya ait nükleotid dizilim bilgisi olmasına rağmen hedef gene ait dizilim bilgisi yer almamaktadır. Bunun yerine hedef gene ait Kyoto Encyclopedia of Genes and Genomes (KEGG) kodu verilmiştir. KEGG veritabanında yapılan sorgular ile TarBase veritabanında yer alan genlere ait dizilim bilgileri elde edilmiştir. Yine TarBase veritabanından çıkarılan etkileşim verileri bir dosya içerisinde derlenerek elde edilen dizilimlerin eşleştirilmesi sağlanmıştır.

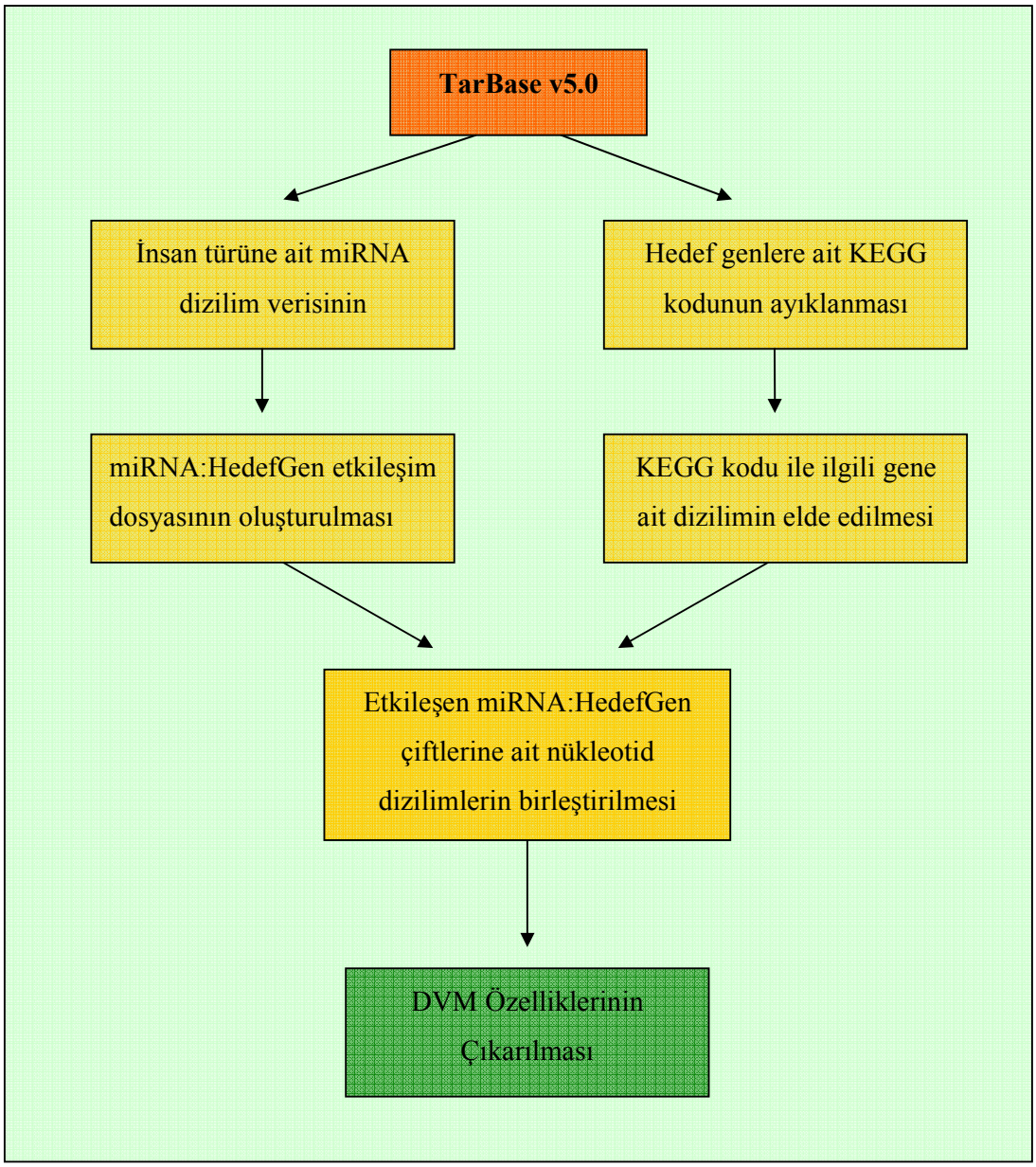
DVM özellikleri yapısal, termodinamik ve pozisyon tabanlı olmak üzere üç grupta toplanmıştır. Yapısal özellikler eşleşmenin olduğu tüm bölgedeki karşılıklı eşleşen veya eşleşmeyen her bir çiftin eşleşme türüne bakılarak hesaplanır. Buna göre (4.2) de verilen özellik kümesi ele alındığında S tüm eşleşmede her bir eşleşmenin türünü gösteren dizi, $\mathfrak{M}$ ise (4.2) deki her bir özelliğin tüm eşleşmede kaç kere tekrar ettiğini gösteren dizidir. Özellik değerleri hesaplanırken bu kümedeki her bir çift için eşleşme bölgesindeki tekrar etme sayısı bulunur (4.3). Daha sonra her bir özellik için tüm eşlemeye olan oran hesaplanır (4.4).

$$F = \{"AA","AC","AG","AU","CA",..,"UA","UC","UG","UU"\} \tag{4.2}$$

$$\sum_{i=1}^{l} (\mathfrak{M}(S(i)) = \mathfrak{M}(S(i)) + 1) \mid S_{1..l}, \mathfrak{M}_{1..16} \tag{4.3}$$

$$\sum_{i=1}^{16} x_i = x_i / l \mid x_i \in \mathfrak{M} \tag{4.4}$$

Yapısal özelliklerin ardından termodinamik özellikler hesaplanır. Bu hesaplama Minimum Free Energy (MFE) algoritmasına göre yapılır. Pozisyon tabanlı özellikler eşleşmede çekirdek bölgesi adı verilen bölge için yapılır. Hesaplamanın yapılabilmesi için öncelikle çekirdek bölgenin tespit edilmesi gerekir. Çekirdek bölge tespit edildikten sonra her bir pozisyon için eşleşmenin türü o özelliğe ait değer olarak alınır.



Şekil 4-2 Veri toplama ve özellik çıkarma süreci

İnsan türüne ait *miR-15a* miRNA'sı ile *HSA:23621* genine ait etkileşim incelenecek olursa pMirna sistemindeki akış şu şekilde olacaktır :

- miR-15a 'ya ait nükleotid diziliminin TarBase'den elde edilmesi

  UAGCAGCACAUAAUGGUUUGUG

- HSA:23621 'e ait nükleotid diziliminin KEGG veritabanından elde edilmesi

  AUGGCCCAAGCCCUGCCCUGGCUCCUGCUGUGGAUGGGCGCGGGAGUGCUGCCUGCCCACGGCACCCAG
  CACGGCAUCCGGCUGCCCCUGCGCAGCGGCCUGGGGGGCGCCCCCCUGGGGCUGCGGCUGCCCCGGGAG
  ACCGACGAAGAGCCCGAGGAGCCCGGCCGGAGGGGCAGCUUUGUGGAGAUGGUGGACAACCUGAGGGGC
  AAGUCGGGGCAGGGCUACUACGUGGAGAUGACCGUGGGCAGCCCCCCGCAGACGCUCAACAUCCUGGUG

GAUACAGGCAGCAGUAACUUUGCAGUGGGUGCUGCCCCCCACCCCUUCCUGCAUCGCUACUACCAGAGG
CAGCUGUCCAGCACAUACCGGGACCUCCGGAAGGGUGUGUAUGUGCCCUACACCCAGGGCAAGUGGGAA
GGGGAGCUGGGCACCGACCUGGUAAGCAUCCCCCAUGGCCCCAACGUCACUGUGCGUGCCAACAUUGCU
GCCAUCACUGAAUCAGACAAGUUCUUCAUCAACGGCUCCAACUGGGAAGGCAUCCUGGGGCUGGCCUAU
GCUGAGAUUGCCAGGCCUGACGACUCCCUGGAGCCUUUCUUUGACUCUCUGGUAAAGCAGACCCACGUU
CCCAACCUCUUCUCCCUGCAGCUUUGUGGUGCUGGCUUCCCCCUCAACCAGUCUGAAGUGCUGGCCUCU
GUCGGAGGGAGCAUGAUCAUUGGAGGUAUCGACCACUCGCUGUACACAGGCAGUCUCUGGUAUACACCC
AUCCGGCGGGAGUGGUAUUAUGAGGUGAUCAUUGUGCGGGUGGAGAUCAAUGGACAGGAUCUGAAAAUG
GACUGCAAGGAGUACAACUAUGACAAGAGCAUUGUGGACAGUGGCACCACCAACCUUCGUUUGCCCAAG
AAAGUGUUUGAAGCUGCAGUCAAAUCCAUCAAGGCAGCCUCCUCCACGGAGAAGUUCCCUGAUGGUUUC
UGGCUAGGAGAGCAGCUGGUGUGCUGGCAAGCAGGCACCACCCCUUGGAACAUUUUCCCAGUCAUCUCA
CUCUACCUAAUGGGUGAGGUUACCAACCAGUCCUUCCGCAUCACCAUCCUUCCGCAGCAAUACCUGCGG
CCAGUGGAAGAUGUGGCCACGUCCCAAGACGACUGUUACAAGUUUGCCAUCUCACAGUCAUCCACGGGC
ACUGUUAUGGGAGCUGUUAUCAUGGAGGGCUUCUACGUUGUCUUUGAUCGGGCCCGAAAACGAAUUGGC
UUUGCUGUCAGCGCUUGCCAUGUGCACGAUGAGUUCAGGACGGCAGCGGUGGAAGGCCCUUUUGUCACC
UUGGACAUGGAAGACUGUGGCUACAACAUUCCACAGACAGAUGAGUCAACCCUCAUGACCAUAGCCUAU
GUCAUGGCUGCCAUCUGCGCCCUCUUCAUGCUGCCACUCUGCCUCAUGGUGUGUCAGUGGCGCUGCCUC
CGCUGCCUGCGCCAGCAGCAUGAUGACUUUGCUGAUGACAUCUCCCUGCUGAAGUGA

- *Dizilimlerin tek dosyada birleştirilmesi*: İlk satırda miRNA dizilimi ikinci satırda hedef gen dizilimi olacak şekilde yapılmalıdır.

- *Eşleşmenin hesaplanması*: Tek bir dosyada birleştirilen iki dizi pMirna tarafından okunur. Eşleşmenin hesaplandıktan sonra sonuç olarak eşleşen bir nükleotid çifti varsa bunlar ilk nükletoid için "(" ve karşılık gelen nükleotid için ")" şeklinde, eğer eşleşmeyen bir nükleotid çifti varsa "." şeklinde ifade edilir. *miR-15a*::*HSA:23621* çifti için sonuç şu şekildedir :

  (((((((. ((((((. (((((((& .))))))) . . .)))))) . . .))))))) .
  1,22 : 1166,1193

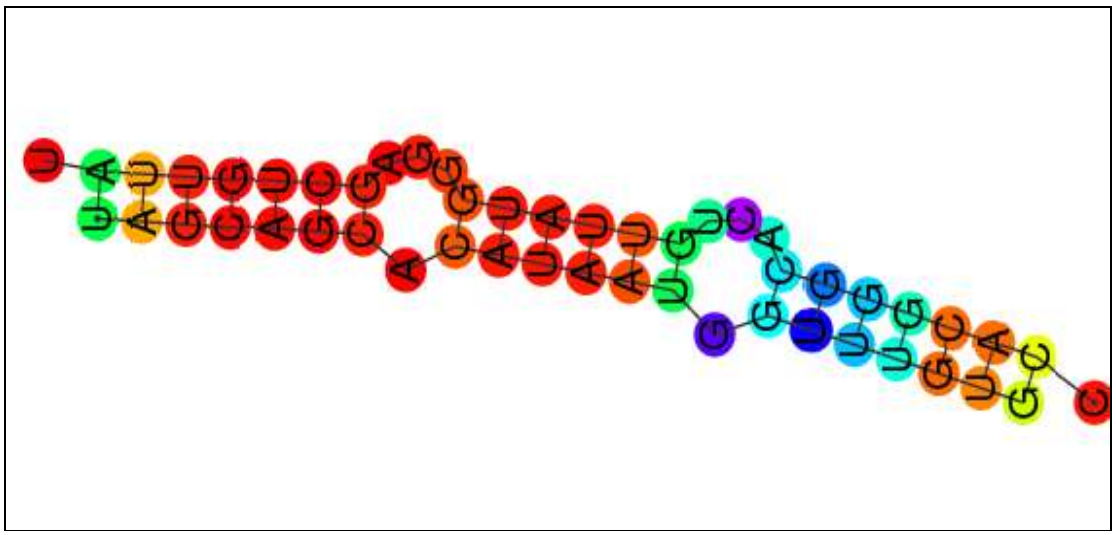
  Bu sonuca göre *miR-15a*'nın tamamı ile *HSA:23621*'in 1166. ile 1193. nükletoidleri arasında bir etkileşim olduğu hesaplanmıştır.

  miR-15a UAGCAGCACAUAAUGGUUUGUG

  HSA:23621 CCACGGGCACUGUUAUGGGAGCUGUUAU

- *Yapısal özelliklerin çıkarılması* : Eşleşmenin olduğu çitflere bakılınca aşağıdaki sonuçlar elde edilir :

$$\begin{array}{ll} \mathfrak{M}(AU)=5 & X(AU)=5/20=0.25 \\ \mathfrak{M}(CG)=3 & X(CG)=3/20=0.15 \\ \mathfrak{M}(GC)=4 & X(GC)=4/20=0.20 \\ \mathfrak{M}(GU)=1 \quad , \; l=20 \quad , & X(GU)=1/20=0.5 \\ \mathfrak{M}(UA)=3 & X(UA)=3/20=0.15 \\ \mathfrak{M}(UG)=4 & X(UG)=4/20=0.20 \end{array}$$



Şekil 4-3 miR-15a::HSA:23621 etkileşimi

- *Termodinamik özelliklerin çıkarılması*: Eşleşmenin tamamına ait minimum enerji hesaplanır. miR-15a::HSA:23621 çifti için bu değer -20.40 olarak hesaplanmıştır.
- *Çekirdek bölgenin pozisyon özelliklerinin çıkarılması* : Çekirdek bölge Şekil 4-3'de kırmızı renkle görülen bölgedir. Bu bölgedeki eşleşmeler yapısal özelliklerin çıkarılması kısmındaki yöntem kullanılarak hesaplanır.

Özellik çıkarma işlemi bittikten sonra elde edilen veriler Çizelge 4-1'deki gibi libsvm öğrenme verisi düzeninde saklanır.[10] Bu düzene göre her satırın başında o satırdaki özelliklerin hangi kümeye ait olduğunu gösteren {+1,-1} ardından ise her bir özelliğe ait numara ve özelliğin sayısal değeri gelmektedir.

Çizelge 4-1 Libsvm öğrenme verisi düzeni

+1	1:0.1234	2:2.3453	3:0.0023	4:34.2131	5:0.000	6:22.2235	....
-1	1:0.000	2:3.000	3:2.1256	4:0.000	5:8.666	6:0.000	....
-1	1:4.012	2:5.555	3:65.00	4:3.3333	5:4.442	6:32.333	....

4.2 Yapısal Özelliklerin Hesaplanması

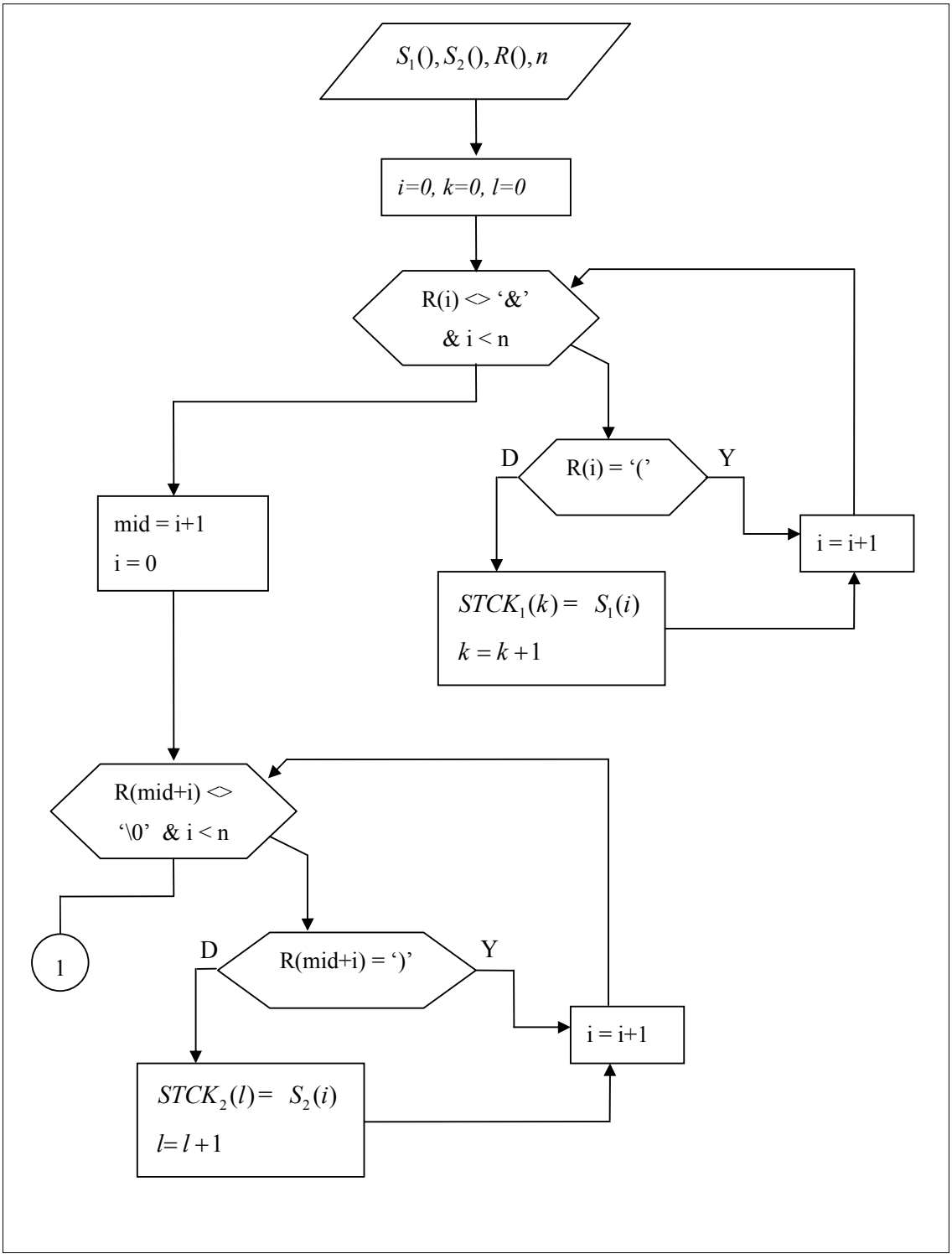
Yapısal özellikler eşleşmenin olduğu bölgeler belirlenmiş iken hesaplanır. $S_1()$ miRNA nükleotid dizilimini, $S_2()$ hedef gene ait nükleotid dizilimini gösteren diziler olsun. Eşleşme bilgisi nokta-parantez düzeninde $R()$ dizisi ile verilsin. Bu düzene göre '(' eşleşmenin olduğu

çiftlerden ilk dizideki nükleotidi, bundan sonra gelen ilk ‘)’ ise bu nükletoide karşılık gelen ikinci dizideki nükleotidi gösterir. ‘&’ birinci dizi için gösterimin bittiğini bundan sonra gelen ifadelerin ikinci dizi için olduğunu gösterir. ‘.’ ise eşleşmenin olmadığı veya yanlış eşleşen çiftleri gösterir. R() dizisindeki ‘(’ ve ‘)’ sayıları eşit olmak zorundadır. Buna göre $S_1()$ ve $S_2()$ dizilerinin elemanları (4.5)’de ifade edildiği gibi {A, C, G, U} kümesine ait elemanlardan oluşur. $R()$ dizisi ise (4.6)’da gösterildiği gibi {‘(’, ‘)’, ‘.’, ‘&’} kümesinin elemanlarından oluşur.

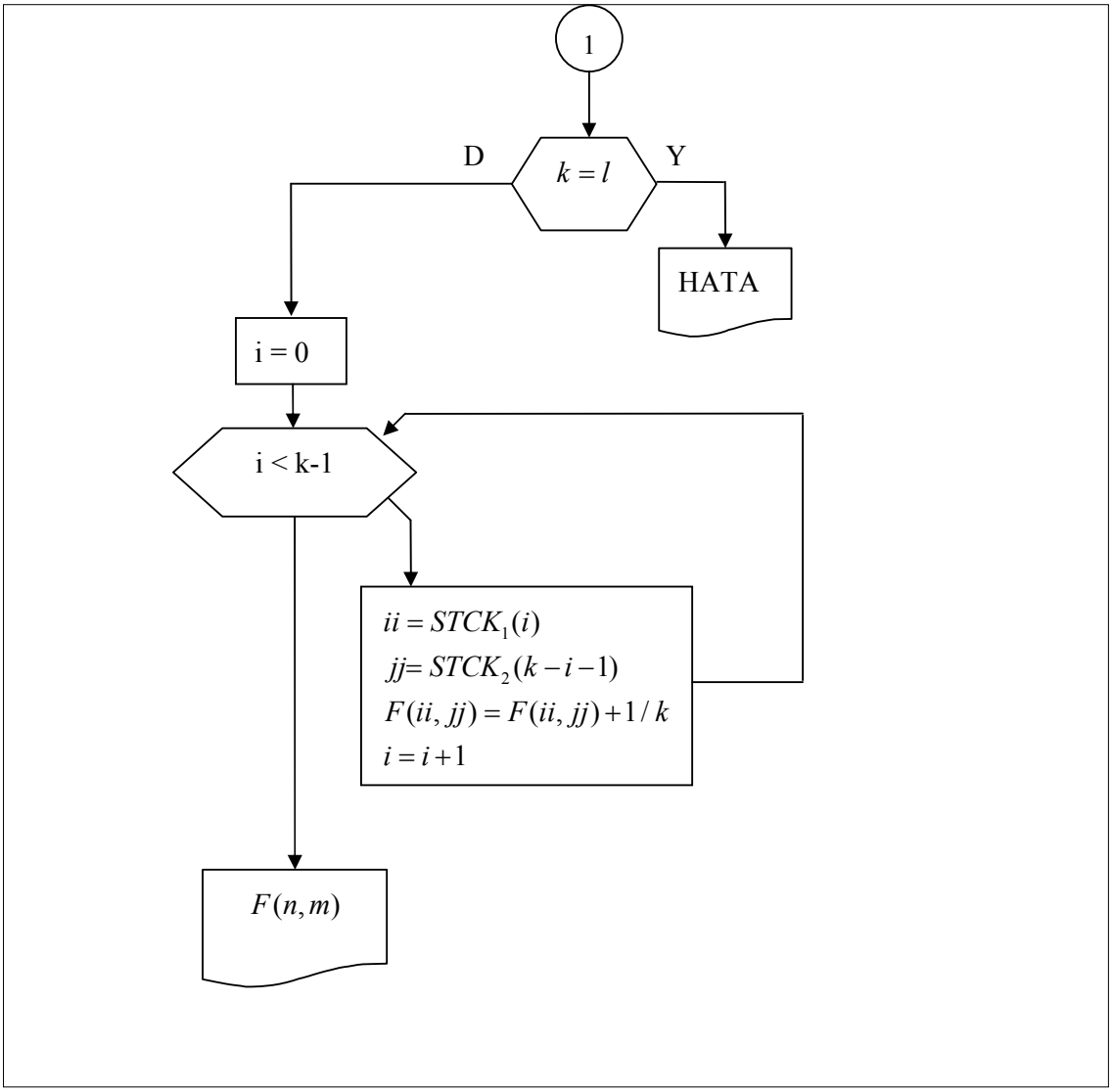
$$\begin{aligned} &\mathbb{N} = \{A, C, G, U\} \\ &S_1(i) \in \mathbb{N}, i = 1..n \\ &S_2(i) \in \mathbb{N}, i = 1..n \end{aligned} \tag{4.5}$$

$$\begin{aligned} &\mathfrak{M} = \{'(', ')', '.', '\&'\} \\ &R(i) \in \mathfrak{M}, i = 1..m \end{aligned} \tag{4.6}$$

Yapısal özellikler eşleşmenin olduğu bölgedeki (4.1)’de gösterilen çiftlerin toplam sayısının bulunması ile hesaplanır. Bulunan değerler toplam eşleşme sayısına bölünerek her bir çiftin eşleşmedeki oranı bulunur. Bu oranlar yapısal özellikleri oluşturur. Bu işi yapan algoritmanın akışı Şekil 4-4 ve Şekil 4-5’de verilmiştir.

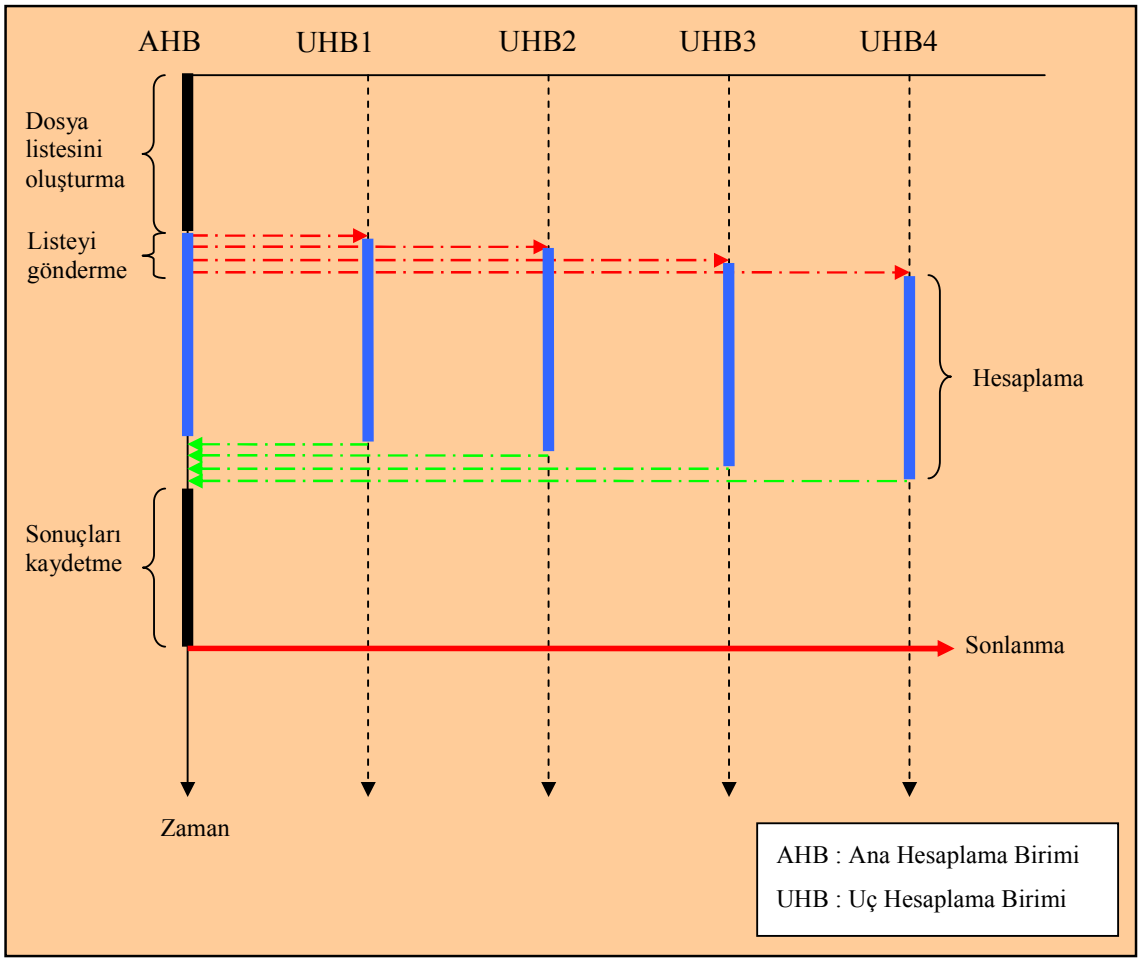


Şekil 4-4 Yapısal özelliklerin hesaplanmasına ilişkin akış



Şekil 4-5 Yapısal özelliklerin hesaplanmasına ilişkin akış -devam

Akışta görüldüğü gibi sonuçta $F(n,m)$ matrisi eşleşen çiftlere ait oranları gösterecek şekilde oluşur. Bu algoritma C programlama dili kullanılarak MPI kütüphaneleri yardımı ile paralel olarak gerçeklenmiştir. Paralelleştirme ilkesi olarak veri bölme yöntemi kullanılmıştır. Bu yönteme göre ana hesaplama birimi ile uç hesaplama birimlerinin paylaşımlı olarak kullandığı bir veri saklama ortamında yer alan veriler hesaplama yapacak birimler arasında paylaştırılır. Her birim kendisine ait veri üzerindeki hesaplamalarını gerçekleştirdikten sonra oluşan özellik verilerini ana hesaplama birimine gönderir. Tüm uç birimlerde çalışan program parçacıkları aynı olduğu için ana birime uç birimlerden gelen verinin düzeni aynıdır. Şekil 4-6'de ana hesaplama birimi ile uç hesaplama birimleri arasındaki haberleşmenin zaman içerisinde ne şekilde gerçekleştiği gösterilmiştir.

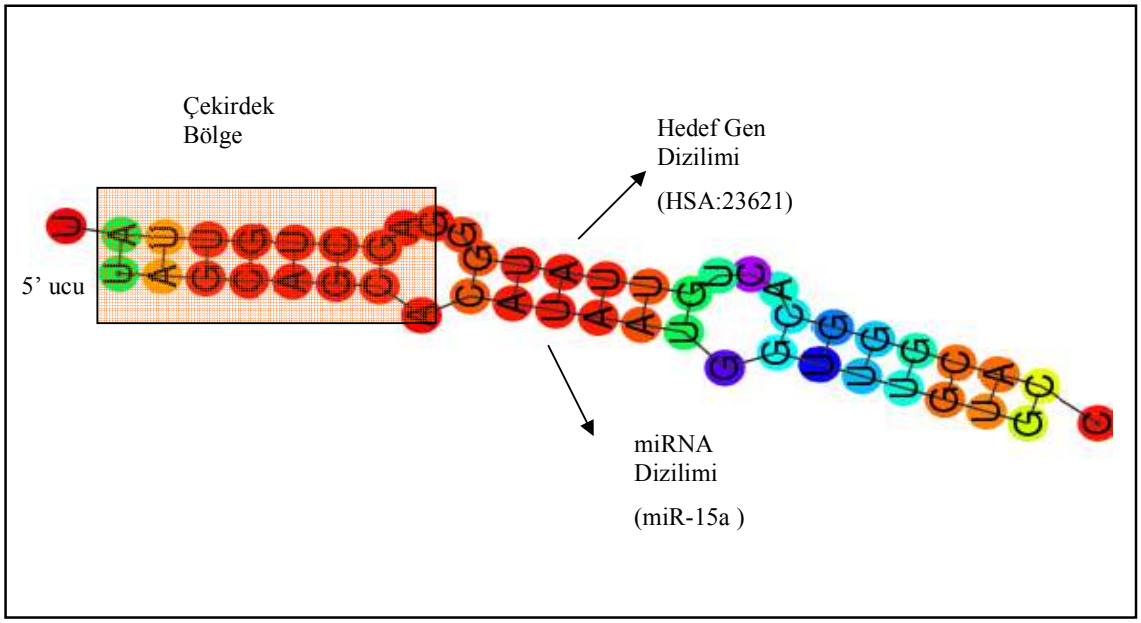


Şekil 4-6 Ana hesaplama birimi ile uç hesaplama birimleri arasındaki iletişim

Hesaplama süreleri her birim için aynı olmayabilir. Ancak pMirna en uygun dağılımı gerçekleştirebilmek için UHB'lere eşit büyüklükte veri gönderir. Bu özellik sayesinde UHB'lerde oluşacak olası bir gecikme yüzünden tüm programın çalışma süresinin uzamasının önüne geçilmesi hedeflenmiştir.

4.3 Çekirdek Bölgesinin Tespiti ve Pozisyon Tabanlı Özelliklerin Hesaplanması

Bir miRNA::HedefGen çitfi için çekirdek bölge ardarda aralıksız veya çok yoğun şekilde 'A-U' ve 'G-C' çiftlerinin bulunduğu bölge olarak ifade edilir. Bu bölgenin tespiti zor olmasa da birden fazla çift için ortak bir özellik belirlemek zorlayıcı olabilmektedir. Zira çekirdek bölgenin uzunluğu ve yeri farklı miRNA'lar için çeşitlilik göstermektedir. Bu yüzden genel kabul görmüş bir kural çerçevesinde pMirna sisteminde miRNA'nın 5' ucundaki ilk 8 çift çekirdek bölge olarak değerlendirilir. Şekil 4-7'de dikdörtgen ile çevrelenmiş bölge miR-15a::HSA:23621 çifti için çekirdek bölgeyi göstermektedir.



Şekil 4-7 miRNA::HedefGen çifti çekirdek bölgesi

Çekirdek bölgenin tespitinden sonra 1'den 8'e her pozisyon için eşleşen çiftler miRNA'dan hedef gene doğru tespit edilir. Yönlü bir tespit olduğu için her pozisyonda 4 farklı eşleşme olasılığı vardır:

$$f = \begin{Bmatrix} 1,\{'A-U'\} \\ 2,\{'U-A'\} \\ 3,\{'G-C'\} \\ 4,\{'C-G'\} \end{Bmatrix}$$

Özellikler hesaplanırken her bir pozisyon için ilgili eşleşme yukarıda verilen sayısal değerler ile ifade edilir. Hesaplama yapısal özelliklerin hesaplandığı süreçte gerçekleşir. Dolayısıyla Şekil 4-6'deki süreç pozisyon tabanlı özelliklerin hesaplanması için de geçerlidir.

4.4 Termodinamik Özelliklerin Hesaplanması

Termodinamik özellik olarak toplam eşleşmenin MFE'ye göre hesaplanan enerji değeri dikkate alınmıştır. Bu değer ViennaRNA paketinin ürettiği sonuçtan doğrudan alınarak işleme konulmaktadır. Her bir çift için yalnızca bir değer hesaplanmaktadır. Şekil 4-6'deki süreç termodinamik özellikllerin hesaplanması için de geçerlidir.

5. DENEYSEL SONUÇLAR

Geliştirilen yöntemler TÜBİTAK ULAKBİM kapsamında yer alan TR-Grid [9] oluşumuna ait bilgisayar kümeleri üzerinde çalıştırılarak sınanmıştır. Başarım ölçütü değişkenleri olarak veri kümesi boyutu ve hesaplama birimi sayısı kullanılmıştır. Veri kümesi olarak 5000, 10.000, 20.000, 50.000, 100.000 ve 250.000 adet nükleotid çifti kullanılmıştır. Bu giriş verileri geliştirilen yöntemler tarafından ayrı ayrı 1, 2, 4, 8, 16, 32 ve 64 hesaplama birimi kullanılarak işlenmiştir. Elde edilen sonuçlar iki farklı başlık altında aşağıda verilmiştir.

5.1 Veri kümesi boyutuna göre karşılaştırmalı sonuçlar

Bu bölümde değişen veri kümesi boyutuna göre okuma, gönderme, hesaplama, alma, kaydetme süreleri ve toplam sürenin artan hesaplama birimi sayısına göre değişimi incelenmiştir. Bahsi geçen okuma süresi, AHB'de giriş verisi olarak verilen dosyanın okunması için harcanan süreyi; gönderme süresi, AHB'den UHB'lere dosya isimlerinin gönderilmesi için harcanan süreyi; hesaplama süresi, her bir hesaplama biriminin ilgili hesaplama için harcadığı ortalama süreyi; alma süresi, hesaplamada oluşan sonuçların UHB'lerden AHB'ye aktarılması için harcanan süreyi; kaydetme süresi, AHB'nin UHB'lerden aldığı sonuçları birleştirip kaydetmek için harcadığı süreyi ve toplam süre de AHB'de tüm programın başlaması ile bitişi arasındaki harcanan süreyi gösterir. Elde edilen sonuçlar Çizelge 5-1,

Çizelge 5-2, Çizelge 5-3, Çizelge 5-4, Çizelge 5-5 ve Çizelge 5-6 ile verilmiştir.Çizelgelerde görülen sayısal değerler saniye cinsinden verilmiştir.

Çizelge 5-1 5.000 tane nükleotid çifti için elde edilen deneysel sonuçlar

HB sayısı	Okuma (sn)	Gönderme (sn)	Hesaplama (sn)	Alma (sn)	Kaydetme (sn)	Toplam (sn)
1	0.010	0.000	544.300	0.000	0.060	544.370
2	0.010	0.040	274.080	0.230	0.070	274.440
4	0.010	0.090	137.380	0.940	0.070	138.490
8	0.010	0.170	66.750	1.170	0.060	68.160
16	0.010	0.430	33.310	1.440	0.060	35.270
32	0.020	1.110	16.650	1.520	0.060	19.470
64	0.010	2.980	8.060	1.450	0.060	13.140

Çizelge 5-2 10.000 tane nükleotid çifti için elde edilen deneysel sonuçlar

HB sayısı	Okuma (sn)	Gönderme (sn)	Hesaplama (sn)	Alma (sn)	Kaydetme (sn)	Toplam (sn)
1	0.020	0.000	1071.490	0.000	0.130	1071.640
2	0.010	0.000	535.080	0.350	0.130	535.580
4	0.010	0.130	267.850	2.570	0.130	270.700
8	0.010	0.430	133.480	4.080	0.120	138.130
16	0.030	1.000	66.840	4.920	0.130	72.940
32	0.020	2.390	33.370	5.390	0.130	41.400
64	0.020	5.710	16.830	4.590	0.140	27.830

Çizelge 5-3 20.000 tane nükleotid çifti için elde edilen deneysel sonuçlar

HB sayısı	Okuma (sn)	Gönderme (sn)	Hesaplama (sn)	Alma (sn)	Kaydetme (sn)	Toplam (sn)
1	0.030	0.000	2122.500	0.000	0.270	2122.800
2	0.040	0.190	1068.910	0.440	0.270	1069.850
4	0.020	0.450	532.020	8.250	0.270	541.010
8	0.030	0.930	264.710	16.460	0.250	282.390
16	0.020	2.130	132.510	18.850	0.270	153.800
32	0.020	4.530	65.980	19.770	0.280	90.730
64	0.020	12.590	32.920	17.270	0.270	63.670

Çizelge 5-4 50.000 tane nükleotid çifti için elde edilen deneysel sonuçlar

HB sayısı	Okuma (sn)	Gönderme (sn)	Hesaplama (sn)	Alma (sn)	Kaydetme (sn)	Toplam (sn)
1	0.180	0.000	5856.800	0.000	0.740	5857.720
2	0.190	0.310	2780.800	2.140	0.720	2784.160
4	0.130	0.590	1366.060	77.000	0.650	1444.440
8	0.130	2.180	689.030	208.520	0.670	900.540
16	0.130	5.250	342.070	224.180	0.630	572.290
32	0.120	12.040	166.740	312.360	0.630	492.010
64	0.070	30.190	82.720	154.070	0.690	268.400

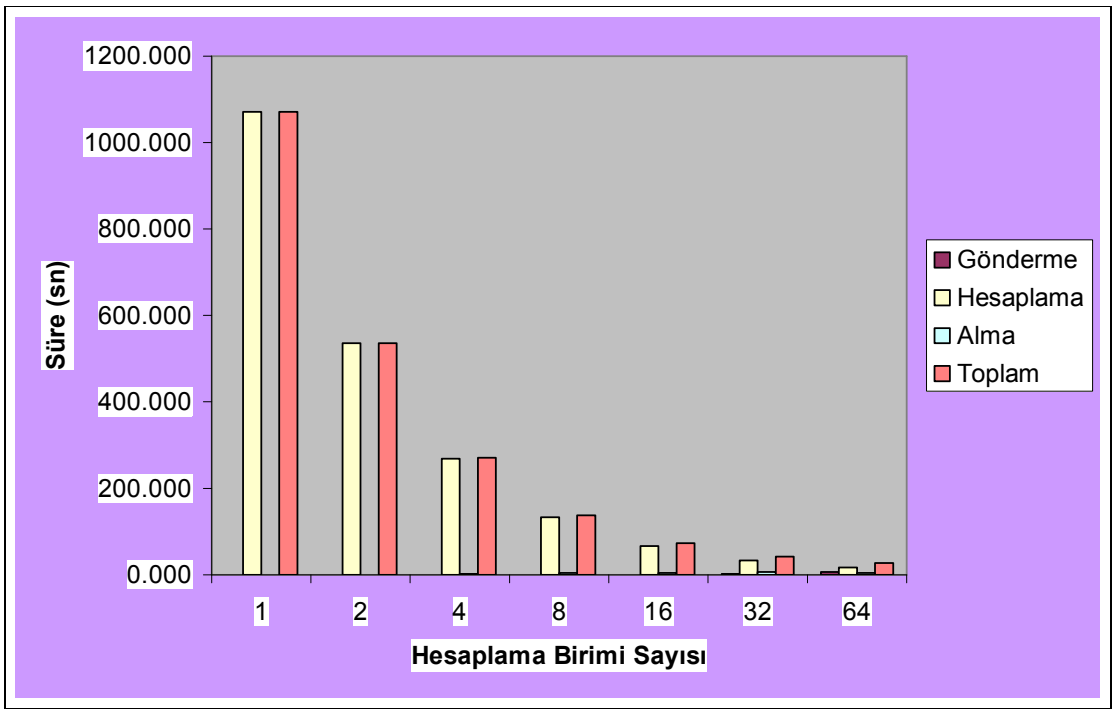
Çizelge 5-5 100.000 tane nükleotid çifti için elde edilen deneysel sonuçlar

HB sayısı	Okuma (sn)	Gönderme (sn)	Hesaplama (sn)	Alma (sn)	Kaydetme (sn)	Toplam (sn)
1	0.120	0.000	10311.620	0.000	1.390	10312.520
2	0.120	0.540	5155.810	1.490	1.390	5159.350
4	0.120	2.430	2653.400	320.650	1.390	2977.990
8	0.170	4.530	1331.710	708.390	1.400	2046.210
16	0.150	9.750	664.310	863.150	1.330	1538.710
32	0.120	23.900	330.950	936.200	1.390	1292.700
64	0.260	61.000	169.740	947.550	1.320	1180.510

Çizelge 5-6 250.000 tane nükleotid çifti için elde edilen deneysel sonuçlar

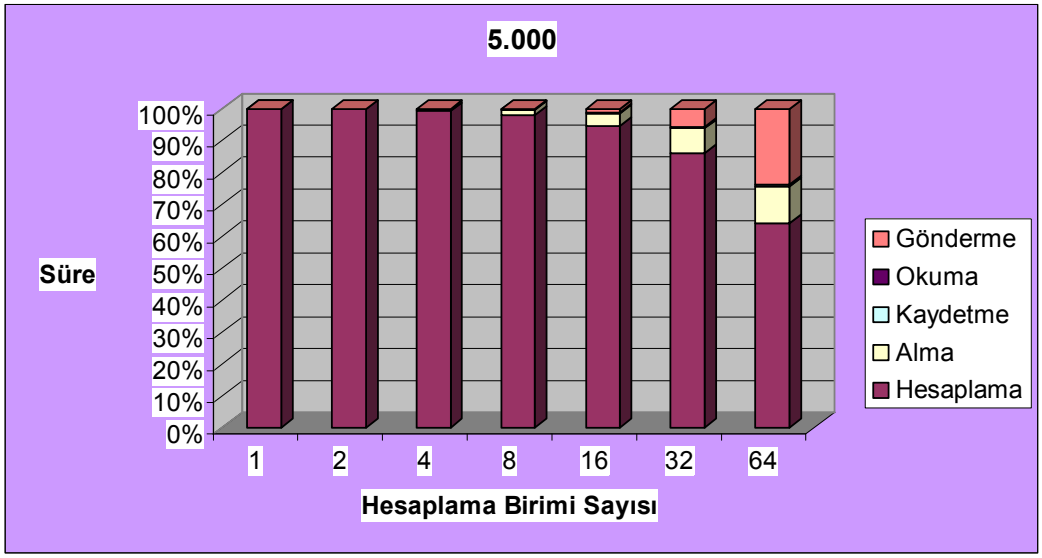
HB sayısı	Okuma (sn)	Gönderme (sn)	Hesaplama (sn)	Alma (sn)	Kaydetme (sn)	Toplam (sn)
1	0.020	0.000	26819.720	0.000	3.190	34002.660
2	0.150	0.930	13409.860	2.250	3.150	16163.240
4	0.510	4.190	6704.930	2614.430	3.410	9327.470
8	1.060	11.200	3408.330	5769.030	3.150	9192.770
16	0.800	26.400	1725.250	7059.200	3.190	8814.860
32	0.580	59.400	837.840	6948.450	3.150	7849.530
64	0.290	156.070	414.040	5751.380	3.470	6325.740

Elde edilen sonuçlar incelendiğinde hesaplama birimi sayısı arttıkça hesaplama süresinin kısaldığı görülmüştür. Şekil 5-1'de görüldüğü gibi 10.000 adet miRNA::HedefGen çifti için kullanılan hesaplama birimi sayısı arttıkça harcanan süre orantılı olarak azalmaktadır. Toplam süre ile hesaplama süresinin birbirine çok yakın olması da verilerin hesaplama birimleri üzerindeki dağılımının dengeli olduğunu gösterir.

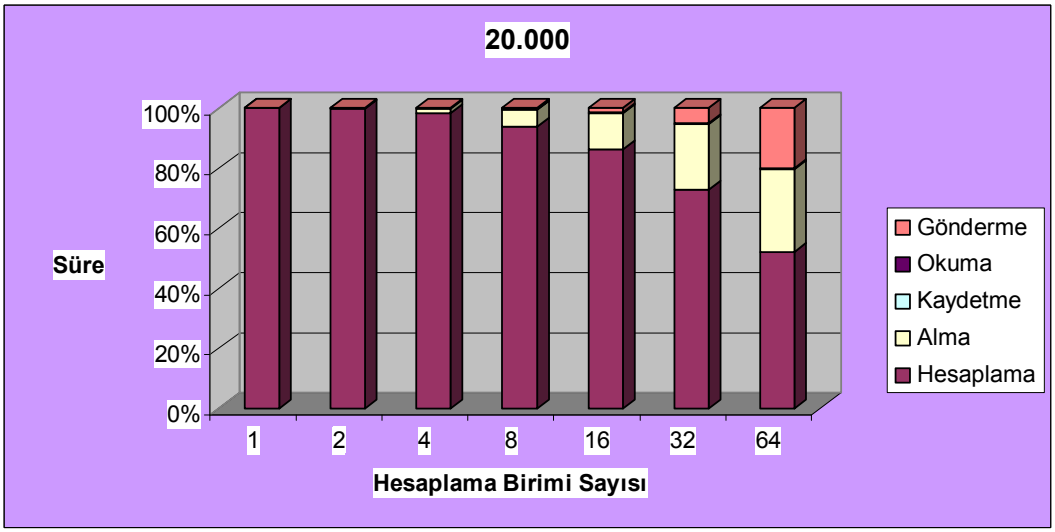


Şekil 5-1 10.000 nükleotid çifti için hesaplama süreleri

Şekil 5-2'de 5.000 miRNA::HedefGen çitfi için yapılan hesaplamalara ait sürelerin dağılımı görülmektedir. Bu şekil incelendiğinde artan hesaplama birimi sayısı ile verinin bu birimlere gönderilmesi için harcanan sürenin de arttığı söylenebilir. Bunun yanında UHB'lerde oluşan sonuçların da AHB'ye gönderilmesi için harcanan sürenin hesaplama birimi sayısı ile doğru orantılı olarak arttığı gözlemlenmektedir. Şekil 5-3 incelendiğinde 20.000 nükleotid çifti için 64 hesaplama birimi kullanıldığı durumda toplam sürenin yarısına yakın bölümünün hesaplama için kullanıldığı; geri kalan sürenin de gönderme-alma işlemleri için kullanıldığı görülmektedir.

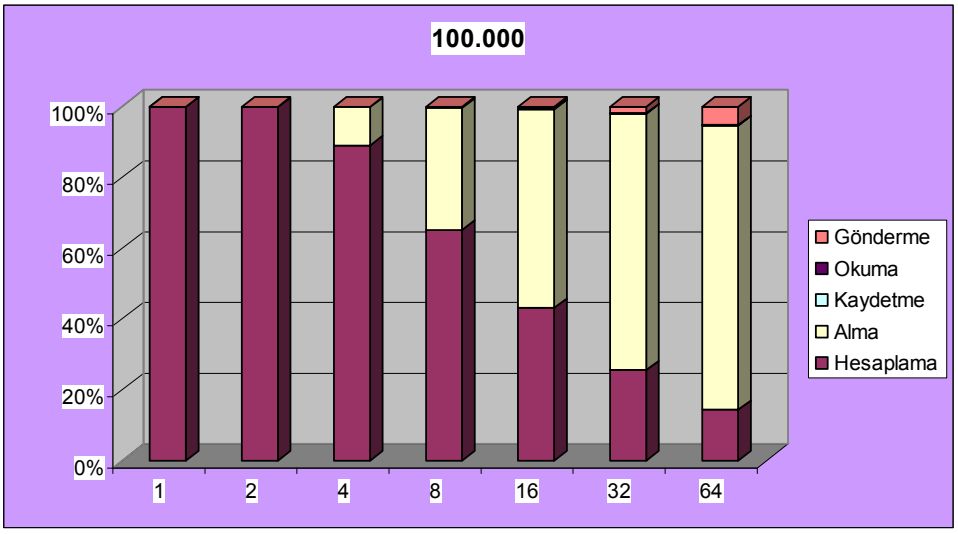


Şekil 5-2 5.000 nükleotid çifti için süre dağılımı



Şekil 5-3 20.000 nükleotid çifti için süre dağılımı

Veri miktarı arttıkça gönderme-alma işlemleri için harcanan sürenin toplam süreye oranı artmaktadır. Şekil 5-4'te 100.000 nükleotid çiftinin hesaplamalarına ait süre dağılımı verilmiştir. Yine 64 hesaplama birimi ile yapılan hesaplamalar incelenecek olursa toplam sürenin %90'ının gönderme-alma işlemleri için harcandığı görülür.



Şekil 5-4 100.000 nükleotid çifti için süre dağılımı

5.2 İşlem türüne göre karşılaştırmalı sonuçlar

İşlem türüne göre karşılaştırmalar her bir işlemin değişen veri miktarı ve hesaplama birimi sayısından nasıl etkilendiğini incelemek amacıyla yapılmıştır. Çizelge 5-7, Çizelge 5-8, Çizelge 5-9, Çizelge 5-10, Çizelge 5-11 ve Çizelge 5-12 her bir işlem için değerleri gösterir.

Çizelge 5-7 Okuma için harcanan süreler

HB sayısı	5000	10000	20000	50000	100000	250000
1	0.010	0.020	0.030	0.180	0.030	0.020
2	0.010	0.010	0.040	0.190	0.120	0.150
4	0.010	0.010	0.020	0.130	0.120	0.510
8	0.010	0.010	0.030	0.130	0.170	1.060
16	0.010	0.030	0.020	0.130	0.150	0.800
32	0.020	0.020	0.020	0.120	0.120	0.580
64	0.010	0.020	0.020	0.070	0.260	0.290

Çizelge 5-8 Gönderme için harcanan süreler

HB sayısı	5000	10000	20000	50000	100000	250000
1	0.000	0.000	0.000	0.000	0.000	0.000
2	0.040	0.000	0.190	0.310	0.540	0.930
4	0.090	0.130	0.450	0.590	2.430	4.190
8	0.170	0.430	0.930	2.180	4.530	11.200
16	0.430	1.000	2.130	5.250	9.750	26.400
32	1.110	2.390	4.530	12.040	23.900	59.400
64	2.980	5.710	12.590	30.190	61.000	156.070

Çizelge 5-9 Hesaplama için harcanan süreler

HB sayısı	5000	10000	20000	50000	100000	250000
1	544.300	1071.490	2122.500	5856.800	10311.620	26819.720
2	274.080	535.080	1068.910	2780.800	5155.810	13409.860
4	137.380	267.850	532.020	1366.060	2653.400	6704.930
8	66.750	133.480	264.710	689.030	1331.710	3408.330
16	33.310	66.840	132.510	342.070	664.310	1725.250
32	16.650	33.370	65.980	166.740	330.950	837.840
64	8.060	16.830	32.920	82.720	169.740	414.040

Çizelge 5-10 Sonuçları almak için harcanan süreler

HB sayısı	5000	10000	20000	50000	100000	250000
1	0.000	0.000	0.000	0.000	0.000	0.000
2	0.230	0.350	0.440	2.140	1.490	2.250
4	0.940	2.570	8.250	77.000	320.650	2614.430
8	1.170	4.080	16.460	208.520	708.390	5769.030
16	1.440	4.920	18.850	224.180	863.150	7059.200
32	1.520	5.390	19.770	312.360	936.200	6948.450
64	1.450	4.590	17.270	154.070	947.550	5751.380

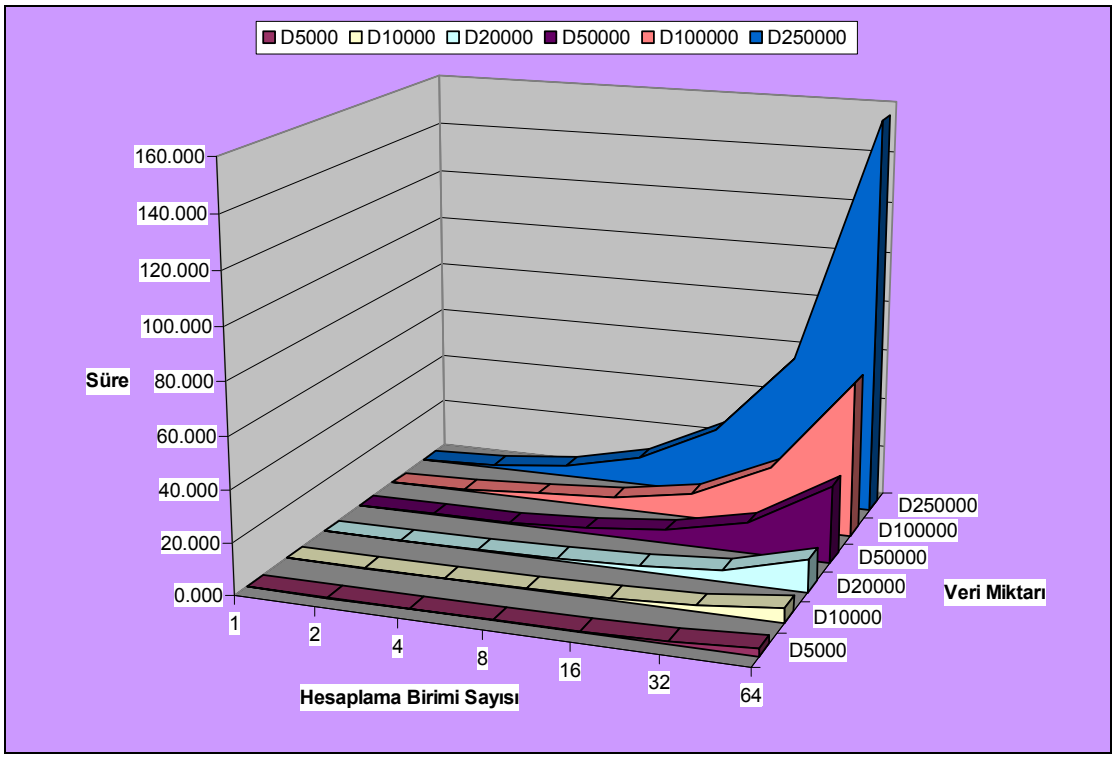
Çizelge 5-11 Sonuçları kaydetme için harcanan süreler

HB sayısı	5000	10000	20000	50000	100000	250000
1	0.060	0.130	0.270	0.740	1.390	3.190
2	0.070	0.130	0.270	0.720	1.390	3.150
4	0.070	0.130	0.270	0.650	1.390	3.410
8	0.060	0.120	0.250	0.670	1.400	3.150
16	0.060	0.130	0.270	0.630	1.330	3.190
32	0.060	0.130	0.280	0.630	1.390	3.150
64	0.060	0.140	0.270	0.690	1.320	3.470

Çizelge 5-12 Toplam süreler

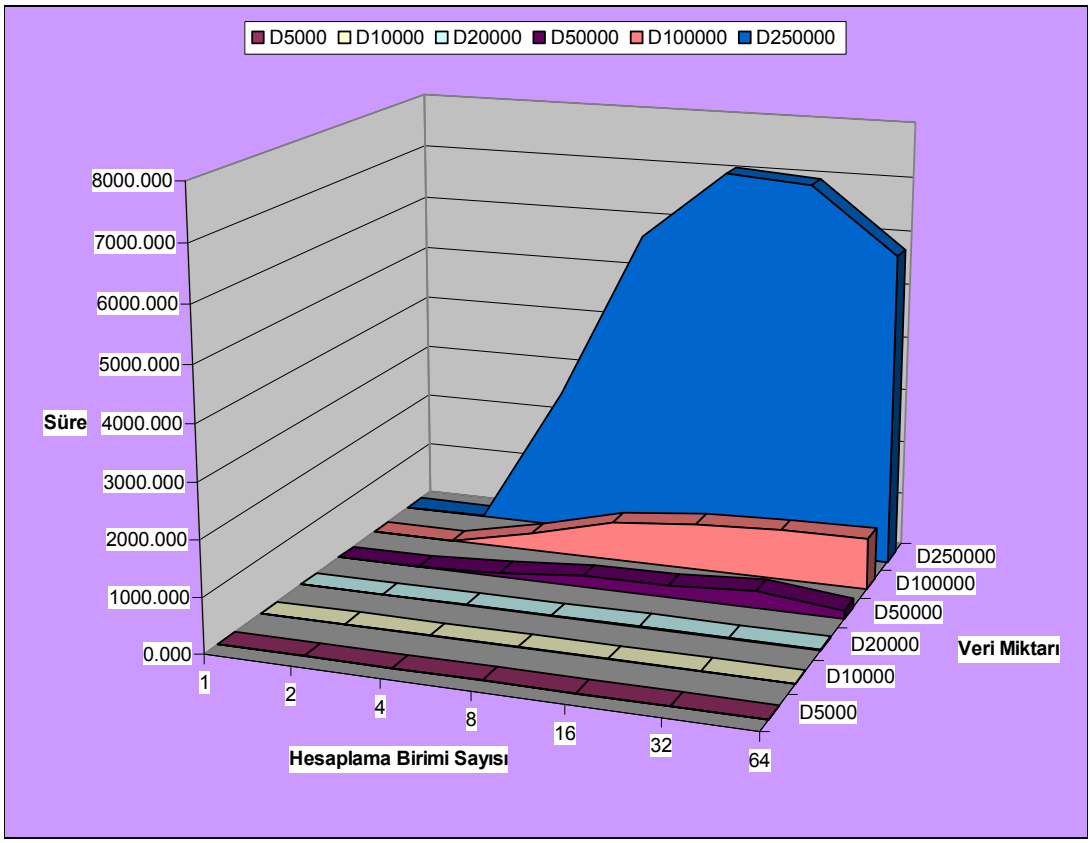
HB sayısı	5000	10000	20000	50000	100000	250000
1	544.370	1071.640	2122.800	5857.720	10853.540	34002.660
2	274.440	535.580	1069.850	2784.160	5159.350	16163.240
4	138.490	270.700	541.010	1444.440	2977.990	9327.470
8	68.160	138.130	282.390	900.540	2046.210	9192.770
16	35.270	72.940	153.800	572.290	1538.710	8814.860
32	19.470	41.400	90.730	492.010	1292.700	7849.530
64	13.140	27.830	63.670	268.400	1180.510	6325.740

İşlem türüne göre karşılaştırmalı çizelgeler incelendiğinde okuma ve kaydetme için harcanan sürelerin hesaplama birimi sayısı ve/veya veri miktarı değiştiğinde çok fazla değişim göstermediği görülmüştür. Buna karşın hesaplama süresi ve gönderme-alma süreleri orantısal değişim göstermiştir. Şekil 5-5'te gönderme işlemine ait yapılan ölçümlerin süreleri yer almaktadır.



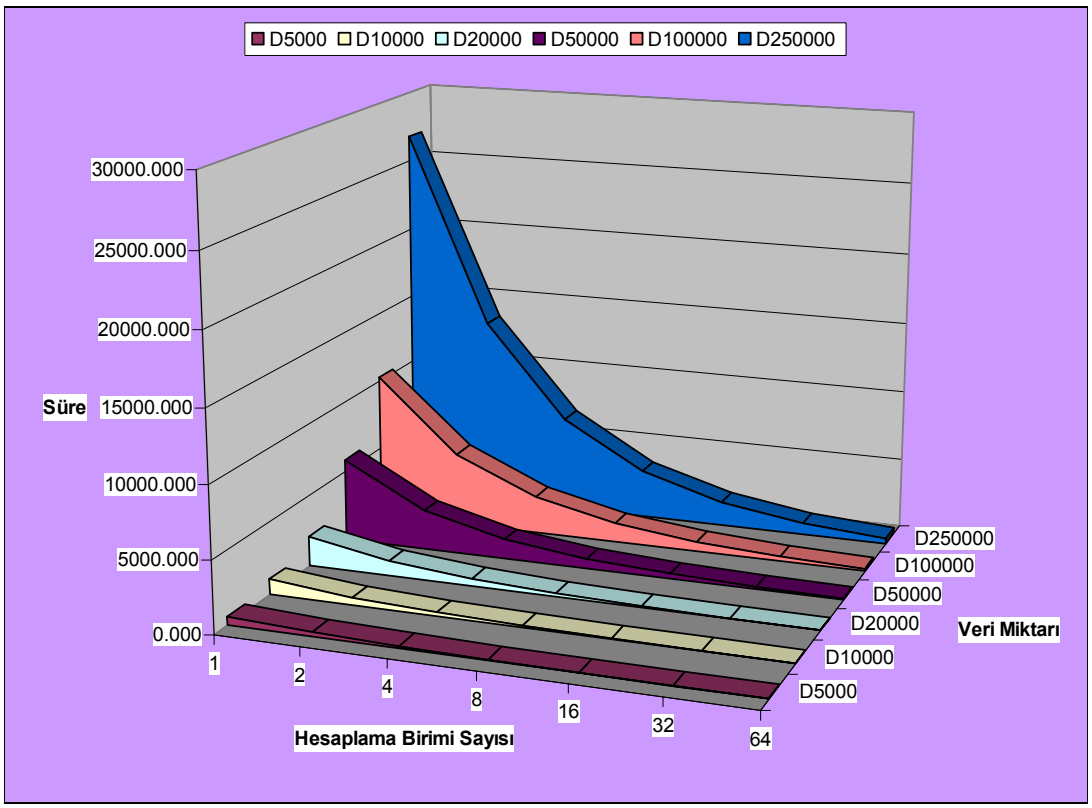
Şekil 5-5 Gönderme işlemine ilişkin süre dağılımı

Buna göre veri gönderilecek hesaplama birimi sayısı arttıkça gönderme süresinin uzadığı aynı zamanda veri miktarı artışının da doğru orantılı olarak bu süreyi uzattığı görülmektedir. Verilerin UHB'lerden toplanması için harcanan süre dağılımı Şekil 5-6'da verilmiştir. Veri miktarı arttıkça sürenin belirgin bir biçimde arttığı görülmektedir. Bunun sebebi artan veri miktarı ile hesaplamalar sonucu oluşan verilerin de miktarının artmasıdır.

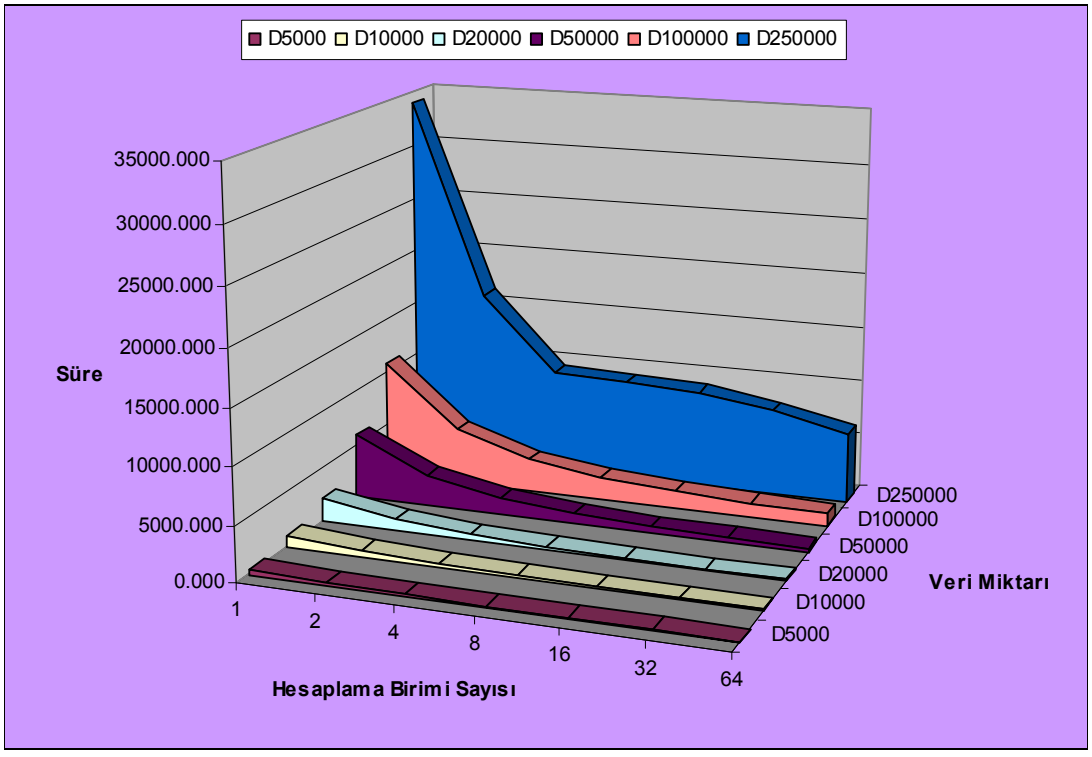


Şekil 5-6 UHB'lerden veri toplama işlemine ilişkin süre dağılımı

Veri miktarındaki artış kaçınılmaz bir şekilde süreç içerisindeki tüm işlemlerin süresini uzatmaktadır. Fakat hesaplama birimi sayısı arttıkça işlemler için harcanan toplam süreler azalmaktadır. Şekil 5-7 UHB'lerde harcanan hesaplama sürelerini, Şekil 5-8 ise toplam çalışma sürelerini göstermektedir. Görüldüğü gibi hesaplama birimi sayısı arttıkça işlem süreleri kısalmaktadır. UHB'lerdeki sürelerde doğrusal bir azalma görülürken, toplam sürelerde hesaplama birimi sayısı arttıkça doğrusal sayılamayacak türde bir azalma gözlenmektedir. Bunun nedeni hesaplama birimi sayısı arttıkça UHB'lerdeki işlem süresi kısalırken UHB'lerde oluşan verilerin AHB'ye gönderilmesi için harcanan sürenin artması ve dolayısıyla da toplam sürenin artmasıdır.



Şekil 5-7 UHB'lerdeki işlem süreleri dağılımı



Şekil 5-8 Toplam süre dağılımı dağılımı

Gönderme süresindeki bu artışlar toplam süreyi ne kadar artırsa da süredeki azalmanın yönünü değiştirecek nitelikte güce sahip değildir. Ancak veri miktarı sabit iken hesaplama birimi sayısındaki artış belli bir noktadan sonra etkisiz hale gelecektir. Zira hesaplama için harcanan süreler sürekli azalırken sonuçların toplanması için harcanan süreler sürekli artacaktır. Bu artmaların ve azalmaların birbirini dengelediği noktada ise hesaplama birimi sayısının artışının toplam süreyi azaltmada etkisi olmayacaktır.

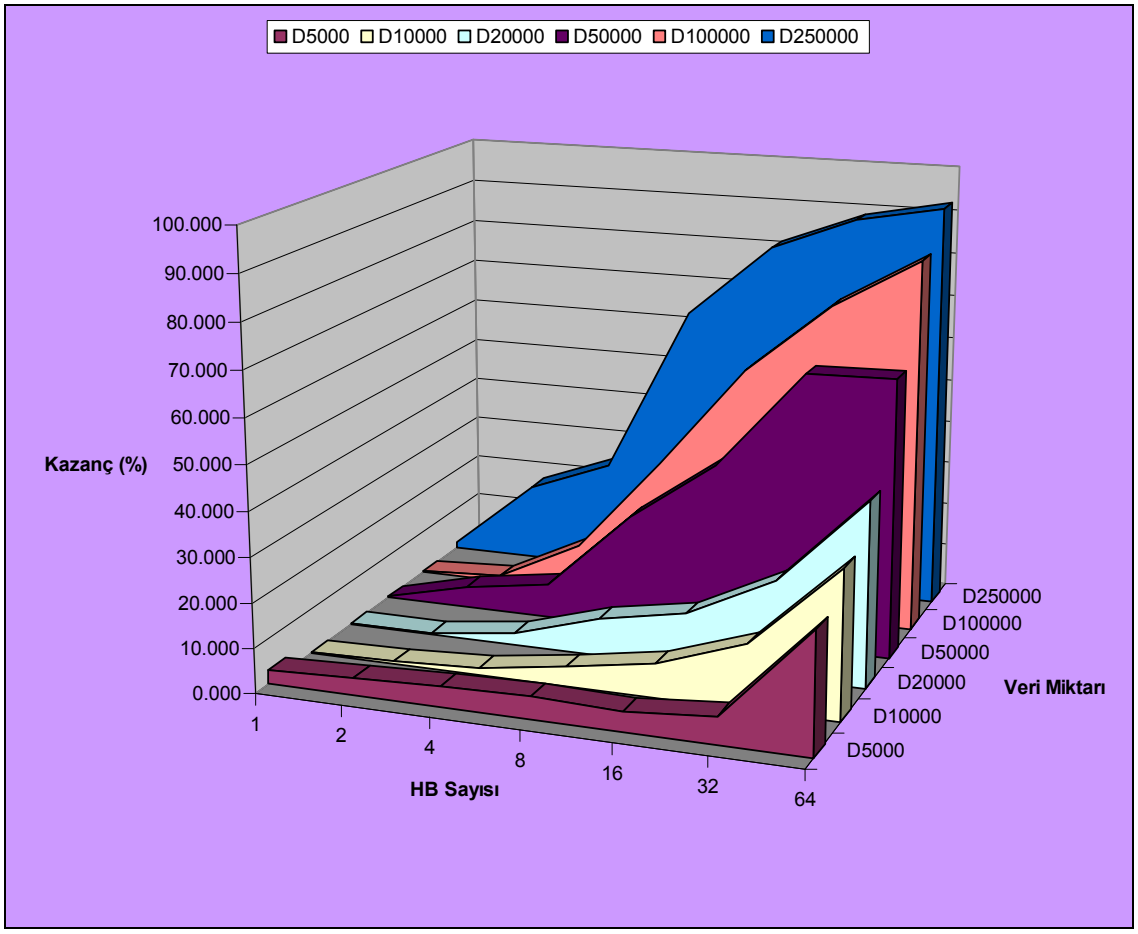
5.3 İyileştirilmiş Yöntem ve Sonuçları

Mevcut yöntem kullanıldığında veri miktarı arttıkça hesaplama sonuçlarının AHB'ye gönderilmesi için harcanan sürenin de arttığı gözlemlenmiştir. Yöntemin daha verimli çalışabilmesi için algoritmada yapılan bir değişiklik ile sonuçların AHB'ye ulaştırılması daha hızlı hale getirilmiştir. Bu değişikliğe göre UHB'ler hesaplamaları yaptıktan sonra sonuçları ağ üzerinden göndermek yerine yüksek hızlı paylaşımlı depolama birimine kaydetmektedir. Tüm hesaplamalar bittiğinde AHB kaydedilen dosyaları birleştirip tek sonuç dosyası haline getirmektedir. İyileştirilmiş yönteme ait başarım sonuçları Çizelge 5-13'te verilmiştir. UHB sayısı arttıkça mesajlaşma arttığından yapılan iyileştirme ile mesajlaşma için harcanan süre ortadan kaldırılmıştır. Çizelge 5-13'te aşağıya ve sağa doğru gittikçe elde edilen kazancın arttığı görülmektedir. Bunun sebebi aşağıya ve sağa gittikçe sırasıyla hesaplama birimi sayısının ve veri miktarının artmasıdır.

Çizelge 5-13 İyileştirilmiş yönteme ait kazanım yüzdeleri

HB Sayısı	5000	10000	20000	50000	100000	250000
1	3.071	0.100	0.260	0.150	0.240	1.340
2	3.768	0.525	0.287	4.834	1.300	17.689
4	4.520	1.515	2.739	7.785	10.844	25.239
8	4.974	4.416	8.605	25.912	32.893	64.117
16	4.446	7.650	12.133	39.962	56.614	81.228
32	6.061	14.444	21.823	62.403	73.068	88.874
64	26.788	33.094	41.542	62.899	84.620	92.714

Şekil 5-9'da iyileştirilmiş yönteme ait karşılaştırmalı kazanç yüzdelerini gösteren grafik verilmiştir.



Şekil 5-9 İyileştirilmiş yönteme ait karşılaştırmalı kazanç yüzdeleri

250.000 örnek için 64 hesaplama birimi kullanıldığı durumda %100'e yakın kazanç sağlanmıştır. Yani toplam hesaplama süresi yarıya inmiştir. Burada dikkat edilmesi gereken nokta şudur : Yapılan iyileştirme UHB'lerdeki hesaplama hızında bir değişime neden olmamıştır. Sadece sonuçların AHB'ye gönderilmesi kısmına olumlu yönde etki etmiş toplam hesaplama sürelerini azaltmıştır.

6. SONUÇLAR VE ÖNERİLER

Bu tez çalışması kapsamında miRNA::HedefGen çiftlerini ait verileri paralel hesaplama yöntemleri kullarak geliştirilmiş DVM uygulamaları ile işleyip miRNA'ların hedef aldığını genleri bulmaya yönelik pMirna adında bir sistem geliştirilmiştir. Çalışmanın hedefi artan veri miktarı ile birlikte uzun hesaplama süresi gerektiren DVM uygulamalarının paralel programlama sayesinde daha kısa sürelerde doğru sonuç vermesini sağlamaktır.

Geliştirilen sistem 3 ana bölümden oluşmaktadır :

1. Veri toplama ve özellik çıkarma
2. Öğrenme
3. Sınama

Çalışmada ağırlıklı olarak özellik çıkarma bölümü üzerinde durulmuştur. miRNA::HedefGen çiftlerine ait özellikler yapısal, termodinamik ve pozisyon tabanlı olmak üzere 3 başlık altında incelenmiştir. Bu özellikler hesaplanırken paralel hesaplama yöntemleri kullanılmış ve hesaplama sürecinde doğrusal bir hızlanma sağlanmıştır. Örneğin 64 işlemci ile yapılan bir denemede işlem süresi 1 işlemci ile yapılan denemeye göre ortalama 62 kat azalmıştır. Bu deneme geliştirilen yöntemin ne kadar etkili sonuçlar verdiğinin göstergesidir.

Veri miktarının artması hesaplama birimleri arasındaki iletişimin de yoğunluğunu artıracağından toplam süreyi olumsuz yönde etkileyebilmektedir. Başarımı olumsuz yönde etkileyen bu durum donanımsal çözümlerle kısmen bertaraf edilebilecek olsa da aradaki iletişim yoğunluğunun azaltılması yönünde geliştirme yapılması daha akılcı olacaktır. Bu çalışmada AHB ile UHB arasındaki iletişimin azaltılmasına yönelik olarak öne sürülen yöntem giriş verilerinin paylaşımlı bir depolama birimi üzerinde tutularak ağ üzerinden değil yüksek hızlı depolama birimleri üzerinden dağıtılmasını sağlamaktır. AHB UHB'lere sadece göndermek istediği veriyi içeren dosyaların isimlerini gönderir. UHB'ler ilgili dosyalara paylaşımlı depolama birimi üzerinden çok yüksek hızda erişir. Bu sayede ağ üzerinde oluşacak yüksek yoğunluktaki iletişimin önüne geçilir.

7. KAYNAKLAR

Mitra, R. ve Bandyopadhyay, S. (2009) ”Improvement of MicroRNA Target Predcition Using An Enhanced Feature Set : A Machine Learning Approach” ,IACC 2009,6-7 Mart 2009 Patiala, India

Liu, H., Yue, D., Zhang, L. ve Huang, Y.F. (2008), “A SVM Based Approach for miRNA Target Prediction”, Proceedings of Seventh International Conference on Machine Learning and Cybernetics, 12-15 Temmuz 2008, Kunming

Kim, S.K., Nam, J.W., Rhee, J.K., Lee, W.J. ve Zhang, B.T. (2005), “miTarget: microRNA target gene prediction using a support vector machine”, BMC Bioinformatics, 1471-2105/7/411

Xue, C., Li, F., He, T., Liu, G.P., Li, Y. ve Zhang, X. (2005), ”Classification of real pseudo microRNA precursors using local structure-sequence features and support vector machine”, BMC Bioinformatics, 1471-2105/6/310

Kim, S.K., Nam, J.W., Lee, W.J. ve Zhang, B.T. (2005), ”A Kernel Method for MicroRNA Target Prediction Using Sensible Data and Position-Based Features”, IEEE, 0-7803-9387-2/05/$20.00

Yousef, M., Jung, S., Showe, L.C. ve Showe, M.K. (2007), “Recursive Cluster Elimination (RCE) for classification and feature selection from gene expression data”, BMC Bioinformatics, 1471-2105/8/144

Pedersen, R.U. (2004), “Using Support Vector Machines for Distributed Machine Learning”, University of Copenhagen, Copenhagen, Denmark. ISSN: 0107-8283

Schölkopf, B., Sung, K., Burges, C., Girosi, F., Niyogi, P., Poggio, T. ve Vapnik, V. (1996), ”Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers”, Massachusetts Institute of Technology Artifical Intelligence Laboratory, C.B.C.L.142

Chang, E.Y., Zhu, K., Wang, H. ve Bai, H. (2008), “Parallelizing Support Vector Machines on Distributed Computers”, Google Research, Beijing, China

Jan, M. ve Zimmerman, O. (2006), “Parallel Cost-Sensitive Support Vector Machine Software for Classification”, John von Neumann Institute for Computing, ISBN-13: 978-3-9810843-0-6

Hsu, C.W., Chang, C.C. ve Lin, C.J. (2008), “A Practical Guide to Support Vector

Classification", Department of Computer Science, National Taiwan University, 2 Ekim 2008, Taiwan

Yom-Tov, E. (2004), "A parallel training algorithm for large scale support vector machines", IBM Haifa Research Labs, Haifa 31905, 28 Kasım 2004, İsrail

Rudner, L.M. (2004), "Expected Classifier Accuracy", Practical Assessment Research & Evaluation, Nisan 2004, San Diego

1. http:// diana.cslab.ece.ntua.gr
2. http://en.wikipedia.org/wiki/File:Svm_separating_hyperplanes.png
3. http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1594580&rendertype=figure&id=F1
4. http://en.wikipedia.org/wiki/File:Svm_max_sep_hyperplane_with_margin.png
5. http://imtech.res.in/raghava/rbpred/svm.jpg
6. http://www.mathworks.com/matlabcentral/fx_files/17204/3/kernel_density.jpg
7. http://mi.eng.cam.ac.uk/~kkc21/thesis_main/img205.gif
8. http://mi.eng.cam.ac.uk/~kkc21/thesis_main/img193.gif
9. http://www.grid.org.tr
10. http://www.csie.ntu.edu.tw/~cjlin/libsvm/

ÖZGEÇMİŞ

Doğum Tarihi	17.01.1983	
Doğum Yeri	Salihli	
Lise	1996 – 2000	Salihli Türkbirliği Lisesi
Lisans	2001 – 2006	Yıldız Teknik Üniversitesi Elektrik-Elektronik Fakültesi Bilgisayar Mühendisliği Bölümü
Yüksek Lisans	2006 – 2009	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Bölümü
Çalıştığı Kurumlar	2006 – 2008 2008 – 2009	YTÜ Bilgi İşlem Merkezi, Uzman Koç.Net A.Ş., Yedekleme Uzmanı
İletişim Bilgileri	Adres	Aziz Mahmut Mh. Tahririye Sk. Birlik Apt. N : 9 D:3 Üsküdar / İstanbul
	Telefon	(+90) 533 544 3822