

19322

İSTANBUL TEKNİK ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

HÜCRESEL NÖRAL AĞLAR VE

UYGULAMALARI

YÜKSEK LİSANS TEZİ

Müh. Sevilay ÖZDEMİR

Tezin Enstitüye Verildiği Tarih : 10 Haziran 1991

Tezin Savunulduğu Tarih : 5 Temmuz 1991

Tez Danışmanı : Prof.Dr. I.Cem GÖKNAR

Diğer Jüri Üyeleri : Prof.Dr. Emre HARMANCI

Doç.Dr. Leyla GÖREN

Yüksel
Dokümantasyon Merkezi

TEMMUZ 1991

Bu tezi hazırlarken gerek konu seçiminde, gerekse çalışmaların yürütmesi esnasında bana olan güven ve desteğini hiçbir zaman kaybetmeyen ve çalışmalarımı yürütmem için bana itici güç oluşturan

Sayın Hocam

Prof. Dr. İ. Cem GÖKNAR'a,
üniversitemizde verdiği seminer ile bu konuyla tanışmamı
ve gönderdiği çok değerli dokümanlarla bu konuda
ilerlememi sağlayan

Sayın

Prof. Tamas ROSKA'ya
gerek çalışmalarım gerekse tezin hazırlanışı esnasında
değerli fikir ve yardımlarından yararlandığım ablam

Sayın

Araş.Gör. İlkay ÖZDEMİR'e,
dokümanların sağlanmasındaki yardımlarından ötürü

Sayın

Araş. Gör. Kenan İSTANBULLU'ya,

Sayın

Araş.Gör. Erkin ÖZMETELER'e

Sayın

Yük. Müh. Burç UZALP'e
teşekkürü bir borç bilirim.

İÇİNDEKİLER

ÖNSÖZ.....	II
ÖZET.....	V
SUMMARY.....	VI
BÖLÜM 1. GİRİŞ.....	1
BÖLÜM 2. NÖRAL AĞLARIN GELİŞİMİ.....	5
2.1 Biyolojik Bulgular.....	7
2.2 Tarihçe.....	10
2.3 Nöromorfik Elektronik Sistemler.....	19
BÖLÜM 3. HÜCRESEL NÖRAL AĞLARIN TANIMI.....	23
3.1 Hücre Yapısı ve Devre Tanımı.....	25
3.2 Küme Kalıbı ve Konvolüsyon Çarpımı.....	28
3.3 Kanonik Hücre Denklemleri.....	30
3.4 Enerji Fonksiyonu ve Denge Durumları.....	34
3.5 Lineer Olamayan ve Gecikmeli Küme Kalıbı Türleri.....	38
BÖLÜM 4. HÜCRESEL NÖRAL AĞLARDA KARARLILIK.....	42
4.1 Karşıt Küme Kalıplı Ağlarda Kararlılık ve Denge Durumlarının Analizi.....	42
4.2 Birlikte Çalışan ve İndirgenemeyen Sistemler.....	48
4.3 Karşıt Olmayan Küme Kalıplarının Kararlılığı.....	51
4.4 Ters İşaretili Küme Kalıplarının Yakınsaklığı.....	54
4.5 Denge Durumlarının Bulunması İçin Bir Algoritma.....	59
BÖLÜM 5. HÜCRESEL NÖRAL AĞLARIN UYGULAMALARI.....	62
5.1 Gürültü Yok Etme.....	64
5.2 Özellik Ayırıştırma Uygulamaları.....	66
5.3 Hücresel Ağlarla Küçük Cisim Sayma.....	79

BÖLÜM 6. HNA YAPISININ İŞARET AYRIŞTIRMA PROBLEMİNE	
UYGULANMASI.....	83
6.1 Problemin Tanımı ve Hopfield Çözümü.....	84
6.2 Hücresel Ağlar ile Gerçekleme.....	86
SONUÇ.....	95
KAYNAKLAR.....	96
ÖZGEÇMİŞ.....	100



ÖZET

Yapay nöral ağlar, son 10 yıldır hızla artan bir ilginin odağı olmaktadır. Günümüzde kullanılan bilgisayarlar digital devreler ile gerçekleştirilmesi çok zor olan bazı tür fonksiyonları insanlar ve hayvanlar kolaylıkla gerçekleştirebilmektedirler. Bilim adamları bir yandan insan beyninin çalışmasını çözmeye çalışırken diğer yandan da edindikleri bilgilerle yeni ve daha güçlü bilgisayarlar, makineler yapmaya çalışmaktadırlar. Bu tezde yapay nöral ağlar hakkında bazı genel bilgiler verildikten sonra nöral ağların özel bir türü olan Hücresel Nöral Ağlar konusu incelenecektir. Hücresel nöral ağlar, nöral ağ teorisindeki bazı boşlukları dolduracak şekilde geliştirilmiş ve görüntü işleme ve tanıma problemlerine başarıyla uygulanmış analog, paralel devrelerdir. HNA'ların teorik yapısı incelendikten sonra çeşitli alanlardaki uygulamalarına da yer verilecektir. Bu tezde hücresel ağlar ile orjinal bir uygulama da gerçekleştirilerek, işaret işlemenin önemli problemlerinden biri olan İşaret Ayırıştırma problemi de yeni bir küme kalıbı bulunarak çözülmüştür.

SUMMARY

CELLULAR NEURAL NETWORKS AND APPLICATIONS

Technology consists of the ideas that people see in the nature and try to copy to ease the life of human beings. Many researchers have taken the examples from nature, animals and humans itself to build up machines, chemical materials, grow plants and even intelligent systems starting from the early ages. Building a machine that can think and behave like a human was always one of the main research issues in the history of mathematics and engineering. Mathematicians, physicians and pshychohologists produced many ideas on thinking machines but until the second half of 20th century they were notbe able to prove their ideas experimentally because of the barriers of technology. Semiconductor and microelectronic technology gave rise to the research on thinking machines and allowed the researchers to carry their ideas from paper to laboratory environment. Then, people realized that conventional circuit and computer theory was not the right answer to solve human-like perception and decision problems. Even the fastest computers of the world were not able to recognize a simple object faster than an animal of high-class. Scientists turned to brain research to find the answer of the question "How does the brain work ?". Several years of brain research showed that the brain was working in a completely different way than our conventional computer electronics which is digital and sequential, brain was analog and parallel. Brain consists of an enormous number of simple cells, called neurons and neurons are working in a highly parallel structure. The elements of a simple neuron can be seen in Fig. 1.

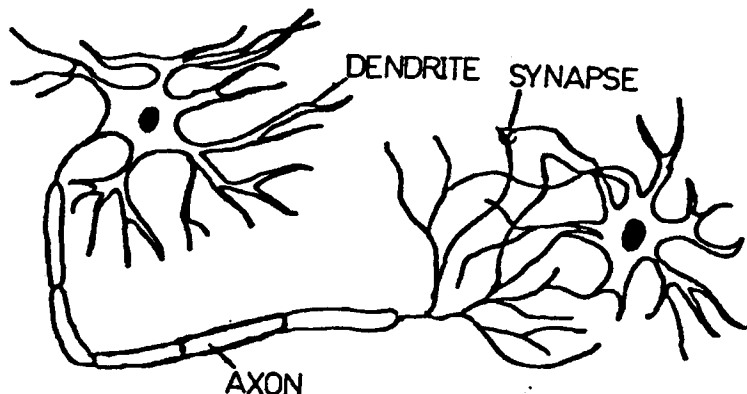


Fig 1. Neuron

A neuron consists of dendrites, the input ports, axons the output ports and synapses forming the connections between neurons. A neuron can be described as a controlled pulse generator which produces a variable frequency pulse train under the controlling inputs of its dendrites. What was surprising that the neurons were slower than an average logic gate but they were producing outputs 100 or even more times faster while working in parallel with other neurons. This fact led the scientists to a new circuit theory, named **Artificial Neural Network Theory**. Artificial neural networks consist of simple analog units with non-linear characteristics. Many researchers have focused on neural networks for nearly 20 years and successfully adapted real-life problems to neural circuits where the conventional computers were inefficient.

Today the term neural network or more properly artificial neural network has come to mean any computing architecture that consists of massively parallel interconnection of simple neural processors. The human nervous system is anatomically subdivided into the central nervous system which comprises the brain and spinal cord and the peripheral nervous system which comprises the spinal and cranial nerves. The brain is subdivided into five major anatomical units called the cerebrum (cortex), cerebellum, midbrain, pons and medulla. Functionally brain can be divided into smaller parts such as motor cortex, visual cortex, auditory cortex, hippocampus and various nuclei groups in the brain system. The average human brain consists of 1.5×10^{10} neurons of various types, with each neuron receiving signals through as many as 10^4 synapses. With that kind of complexity, human brain is considered to be the most complex piece of machinery on earth.

Neurobiologists have taken the bottom-up approach by studying the stimulus-response characteristics of single neurons and networks of neurons. On the other hand, psychologists have taken the top-down approach by studying brain function from the cognitive and behavioral level. Engineers seek solutions to problems that are difficult with today's digital computing technology, problems that are easily solved by people and animals. Therefore they keep an eye on the brain studies and the other eye on making use of available models and paradigms and possibly developing new ones. Engineers try to build more brain-like computers out of neuron-like parts. It was this view that led to the development of many of the earlier models of neurons and neural networks. McCulloch and Pitts in the 1940s showed that the neuron can be modeled as a simple threshold device to perform logic function. Their work was very significant because of showing that the simple

units can perform complex functions when working in parallel.

By the late 1950s and early 1960s neuron models were further defined into Rosenblatt's Perceptron, Widrow and Hoff's Adaline and Steinbuch's Learning matrix. The perceptron received considerable excitement when it was first introduced because of its conceptual simplicity. However was short lived when Minsky and Papert proved mathematically that it can not be used to realize the complex logic functions which are not linearly separable. On the other hand, the faith of adaline was quite different. Because of its linear and adaptive nature, the technic has developed into a powerful tool for adaptive signal processing.

The present interest on neural network research is due inpart to the paper Hopfield published in 1982. In that paper he presented a model of neural computation that was based on the interaction of neurons. The model consisted of a set of first order (non-linear) differantial equations that minimize a certain energy function. The 3 articles of Hopfield and Tank, started a new generation of neural networks research which led the world to a new kind of thinking.

The main advantages of neural networks over conventional computer are their adaptation and learning capability and of course their speed. Furthermore because of their non-linear nature, neural networks can perform functional approximation and signal filtering operations which are beyond optimal linear technics. Neural networks can be used in pattern classification by defining non-linear regions in feature space.

Like all engineering phenomena, neural networks also have their limits. Their massively parallel nature prevents the use of discrete implementation technics because of immediately growing number of connections when the number of neurons is increased. In VLSI implementations, scientists can try to reduce the size of neurons but connections are still covering most of the silicon surface. Wafer-scale integration is another alternative where the scientists try to eliminate the termal effects of large networks.

Cellular Neural Networks are a new generation of neural networks introduced by Chua and Yang at 1988. Their aim was to design a neural network structure which is close to human nervous system and eliminates the disadvantages encountered in VLSI realizations. The main difference of cellular neural networks is their nearest neighborhood connection structure as compared to a fully connected Hopfield Net. In a conventional

cellular neural network, neurons are placed in a two dimensional rectangular grid although the theory can be expanded to higher dimensions. Neurons are only connected to their nearest neighbors in the network matrix and r -Neighborhood can be defined for a cell $C(i,j)$ as follows:

$$N_r(C(i,j)) = \left[C(k,l) \mid \max \{ |k-i|, |l-j| \} \leq r \right]$$

$$\begin{aligned} 1 &\leq k \leq M \\ 1 &\leq l \leq N \\ r &\in \mathbb{N}^+ \end{aligned}$$

This approach reduces the silicon surface used by connections between neurons and as well as the power consumption. Brain also has a similar structure of connection between neurons. An average number of 1000 neurons are in close connection which we call **microscopic connections** and these groups of neurons are also connected to other groups through long synapses that we call **macroscopic connections**.

The internal structure of a cell can be seen in Figure 2. A typical cell in a cellular neural network

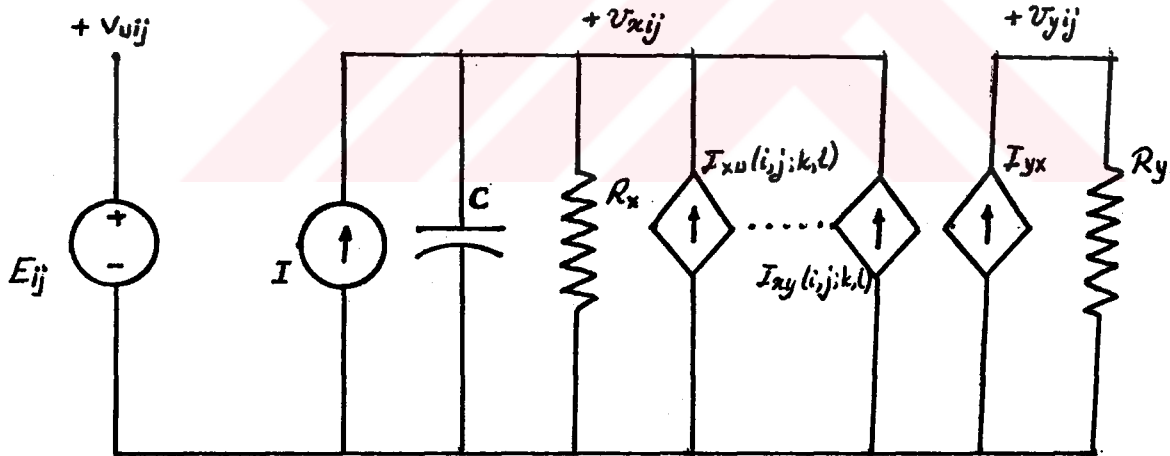


Fig 2. Typical Cell Structure

has 3 nodes u , x and y corresponding to input, state and output of the cell respectively. Cell input is a constant voltage source E_{ij} which is supplied externally. The cell has the following components: A constant current source I , a linear capacitor C , two linear resistors R_x and R_y , numerous voltage controlled current sources which represent the effect of the nearest neighbor neurons and a piecewise-linear voltage controlled current source. Connections with the

neighbors and the inputs are made through weighted currents of the following type :

$$I_{xy}(i,j;k,l) = A(i,j;k,l)v_{ykl} \quad \forall C(k,l) \in N_r(i,j)$$

$$I_{xu}(i,j;k,l) = B(i,j;k,l)v_{ukl} \quad \forall C(k,l) \in N_r(i,j)$$

The only non-linear element of the cell is the voltage controlled current source with the following definition:

$$I_{yx} = \frac{1}{R_y} \cdot f(v_{xij})$$

$$f(v) = \frac{1}{2} \left[|v + 1| - |v - 1| \right]$$

Cellular neural networks can be programmed to perform several tasks through their neighborhood connection weights which are called **cloning templates**. A cloning template is a space invariant matrix that shows the connection weights of the cells in a neighborhood. Connection weights are twofold: The matrix A that holds the weights for output voltages of neighbor cells, which is also called the **feedback operator** and the matrix B that holds the weights for input voltages of neighbor cells, the **control operator**. We can program a cellular neural network with A and B matrices and constant current I, usually these 3 are called a cloning template or analog algorithm of the network.

The theory of cellular neural network depends on the dynamic routes of the individual cells. The cell state equations and the circuit constraints can be given as follows :

$$C \frac{dv_{xij}(t)}{dt} = - \frac{1}{R_x} v_{xij}(t) + A * v_{yij}(t) +$$

$$+ B * v_{uij} + I$$

$$v_{yij}(t) = \frac{1}{2} \left[|v_{xij}(t) + 1| - |v_{xij}(t) - 1| \right]$$

$$v_{uij} = E_{ij} \quad \begin{matrix} 1 \leq i \leq M \\ 1 \leq j \leq N \end{matrix}$$

$$|v_{xij}(0)| \leq 1$$

$$|v_{uij}| \leq 1$$

$$A(i,j;k,l) = A(k,l;i,j)$$

$$C > 0, \quad R_x > 0$$

When cellular neural network theory was first introduced, the elements of cloning templates were all constant real numbers and the stability was dependent on the reciprocity of cloning templates. A cloning template is called to be **reciprocal** iff any two neuron are connected to each other by the same weight. To state mathematically, we can write

$$A(i,j;k,l) = A(k,l;i,j)$$

First stability analysis was depending on that feature and on showing that the energy function

$$\begin{aligned} E(t) = & -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{(k,l) \in N_r(i,j)} A(i,j;k,l) v_{yij}(t) v_{ykl}(t) \\ & + \frac{1}{2R_x} \sum_{i=1}^N \sum_{j=1}^M v_{yij}^2(t) - \sum_{i=1}^N \sum_{j=1}^M \sum_{(k,l) \in N_r(i,j)} B(i,j;k,l) v_{yij}(t) v_{ukl} \\ & - \sum_{i=1}^N \sum_{j=1}^M I v_{yij}(t) \end{aligned}$$

is a Liapunov function for the network. In 1990, Chua and Roska proved that a more relaxed condition of having positive or negative cell linking templates would be enough to guarantee complete stability. At the same time non-linear and delay type templates were introduced by the same authors and some examples given for silicon retina in a following paper. Cellular neural networks seem to be very promising to become analog computing arrays with cloning templates as analog macros whereas analog algorithms will consist of the application of several cloning templates sequentially.

Cellular neural networks are applied to image processing problems with great success. Data corresponding to an image is given as the initial state or is applied to the inputs of the cellular neural network where each neuron corresponds to a pixel of the image. Then using different types of cloning templates, we can extract several features of images independent of the image itself. Many cloning templates for feature extraction can be found in the literature like corner and edge detection, image inversion, shadow detection, closed component detection, hole filling etc. Besides single image processing, cellular neural networks have been applied to some intelligent tasks based on image processing, like small object counting and motion detection. Although some research have been done on the subject of finding the proper cloning template for a given task, no analytical or algorithmic results have been obtained. Most of the cloning templates are found experimentally and their effect on the network's operation is not well explained.

As an original contribution of the thesis, cellular neural network theory is applied to a different area of signal processing, signal decomposition problem which can be defined as follows: Given a sampled function with samples at N different time points, find the components of the original function in terms of Gaussian functions. The shape of Gaussian functions allows us to use the nearest neighbor feature of CNN and the necessary cloning template is only dependent on the sampling period and independent of the function itself in contrast to the solution of Hopfield which was originally suggested with a Hopfield Net. The main advantage of using cellular neural networks for this problem is their fast response which is independent of the networks size and is limited to few microseconds.

BÖLÜM I

GİRİŞ

20. yy teknolojisindeki başdöndürücü gelişmeler, son yıllarda bilim adamlarının insana ve insan davranışlarına olan ilgisini körüklemekte, aynı zamanda da düşüncelerini uygulamaya dökme imkanı tanımaktadır. Sinir ağları, insan ve yüksek sınıftan diğer canlılarda bulunan yaşamsal mekanizmanın temelidir. Düşünme gibi sadece insan türünde bulunan yeteneklerin dışında sinir ağları, canlının fiziksel özelliklerini, kimyasal dengesini, savunma mekanizmalarını vs de oluşturur. Böylesine yoğun ve karmaşık kontrol işlemleri yürüten sinirlerin iç yapısına baktığımızda ise günümüzdeki yarıiletken elektronik devre elemanların çok daha yavaş bir nöron hücresi görürüz. İşte bilim adamları son 10 yıl içerisinde, bu basit ve nispeten yavaş hücreciklerin bir araya geldiklerinde nasıl olup da dünyanın en hızlı bilgisayarından daha hızlı işlem yapabildiklerini araştırmaktadırlar. Araştırmalar insanları nöronların basit modellerini yapıp, bu modeller ile oluşturdukları devrelere insansı bazı işlemleri yaptırmaya çalışmaya götürmüştür. Nöronun modellenmesi, ele alınan örneklere bağlı olarak çok farklı şekillerde olabilmektedir. Örneğin öğrenme yeteneği olan bir nöron yanında, öğrenme yeteneği olmayan fakat çok hızlı cevap verebilen bir nöron modeli yapmak da mümkündür. Yapay sinir ağları çalışmalarının yoğunlaştığı iki büyük alandan bir tanesi öğrenme diğeri ise tanımadır. Öğrenmede devrenin bilgi saklama kapasitesi en önemli rolü oynarken, algılamada giriş verilerindeki bozulmalara karşı dayanıklılık, ortamdan bağımsızlık ve hız daha fazla önem kazanmaktadır. İşlev olarak devre gerçekleştirildikten sonra uygulamaya dönük bazı

optimizasyonlar da yapılır. VLSI teknolojisi ile kolaylıkla gerçekleştirilebilen sinir ağları, diğerlerine göre avantajlıdır çünkü nöronlar arasında hızla artan bağlantı sayısı nedeniyle 4-6'dan fazla nöron içeren devreleri ayırık elemanlarla yapmak olurlu değildir. Bağlantı kolaylığının yanında, devrenin az güç harcaması da aranan bir özelliktir.

Hücresel nöral ağ modeli 1988 yılında L.Chua ve L. Yang tarafından ortaya atılmıştır. Pek çok nöron ve ağ modelinden biri olan hücresel ağların en önemli özelliği, VLSI uygulamaları baz alınarak tasarlanmış olması ve insan doğasına uygunluğudur. Hücresel ağlarda, Hopfield devrelerinde olduğu gibi tüm nöronları doğrudan birbirine bağlamak yerine, insan beynindeki gibi bölgesel bağlantılar kullanılmıştır. Her nöron ancak en yakın komşularına doğrudan bağlı olduğundan hem devrenin kapladığı yer azalmakta, hem de veriler devre üzerinde fazla taşınmadığı için toplam güç harcaması düşük olmaktadır. Hücresel ağların ilk uygulaması el yazması Japon karakterlerinin tanınması olmuştur. Temelde çok karışık şekillerden oluşan Japon alfabesinin aynı zamanda elle yazılması ve harflerde meydana gelebilecek deformasyonların ölçüsü, bu problemin günümüz klasik bilgisayar mimarisiyle çözölemeyeceğini göstermiştir. El yazması harflerden bazı özelliklerin ayrıştırılması ve gene bir eğitilmiş nöral ağ ile elde edilen özelliklere en uygun harfin bulunması üzerine kurulu olan bu sistem, tanıma yüzdesi olarak çok iyi sonuçlar vermiştir. Hücresel ağların önemli özelliklerinden birisi de kararlı bir sonuca yakınsama süresinin devrenin boyutundan bağımsız olması ve bir nöron gecikmesi ile belirlenmesidir. Yakınsama süresinin yaklaşık sabit olması hücresel ağların gelecekte de pek çok uygulama alanı bulacağını göstermektedir. Hücresel ağ teorisi günümüze kadar büyük çoğunlukla görüntü işleme alanında kullanılmıştır. Fakat esnek ve gelişmeye açık yapıları nedeniyle nöral ağ modelleri arasında oldukça umut verici bir konumdadır.

Bu tezde genel sinir ağı özelliklerinden bahsedildikten sonra hücresel ağlar üzerinde detaylı teorik ve pratik bilgi verilmiştir. Ayrıca orjinal bir uygulama da tezde yer almaktadır. 2. Bölümde genel nöral ağ teorisi ve nöral ağların tarih içindeki gelişimi ele alınmıştır. Teknik literatüre geçmiş önemli yayınlardan örneklerin de verildiği bu bölümde kısaca sinir ağlarının öğrenme, çıkarsama yapma, optimizasyon yapma gibi özelliklerinden bahsedilmiştir. VLSI uygulamalarda karşılaşılan güçlükler ve gelecekteki yönelimler de gene bu bölümde ele alınmıştır.

3. Bölüm ise hücresel sinir ağları teorisine giriş yapısındadır. Nöron tanımının, devre yapısının, kullanılan terminolojinin ve nöron iç yapısındaki elemanların özelliklerinin tanıtıldığı bu bölümde ayrıca devre denklemleri ve bazı teoremlere de yer verilmiştir. Gelecekte devre elemanları üzerinde yapılabilecek değişiklikler ve bu değişiklikler sonucunda elde edilecek yeni özellikler de gene bu bölümde ele alınmıştır.

4. Bölümde hücresel ağların kararlılık analizi yer almaktadır. Yapılan her elektronik devrede olduğu gibi, bir devrenin çıkışlarının sonlu ve kararlı olduğunun matematiksel olarak garanti edilmesi gerekmektedir. Hücresel ağlarda kararlılığın var olması için gerek ve yeter koşullar, çeşitli kararlılık analizleri ve devre parametrelerine getirilen kısıtlamalar bu bölümde bulunabilir. 1988 yılından bu yana yapılan tüm kararlılık analiz ve kriterleri ele alınarak birbirleriyle karşılaştırılmıştır. Ayrıca kararlı denge durumlarının bulunması için de bir algoritma verilmiştir.

5. Bölüm hücresel ağların uygulamalarını içermektedir. Özellikle görüntü işleme konusunda yapılmış olan uygulamalar, ayrıntıları ve şekilleri ile verilmiştir. 1988 yılından bu yana yayınlanan tüm hücresel ağ görüntü işleme uygulamaları bu bölümde bulunabilir. Ayrıca 1990 yılında gerçekleştirilen ve temeli görüntü işleme olmasına

karşın başka işlevleri olan nöral devre modellerinden de bahsedilmiştir. Görüntü işlemenin yanında hareket sezme, hız ölçme gibi değişik bazı uygulamalardan da bahsedilmiştir. Bu bölümdeki simülasyonlar Macaristan Bilimler Akademisi'nde Prof. Tamas Roska yönetiminde geliştirilen CNN adlı simulator programı ile hazırlanmıştır. CNN'in teknik ayrıntıları ve referansları da ekler arasında bulunabilir.

6. Bölümde ise bu tez kapsamında hücresel ağlarla gerçekleştirilen özel bir uygulama yer almaktadır. J.Hopfield tarafından 1986 yılında yapılan bir çalışmadan esinlenerek, aynı probleme hücresel ağ mimarisi ile daha gelişmiş bir çözüm getirilmiştir. N adet noktada örneklenmiş bir fonksiyonun Gauss bileşenlerinin bulunmasını sağlayan bu devre yapısı ile, yeni bir hücresel ağ küme kalıbı bulunmuş ve CNN simülatörü yardımıyla çeşitli örnekler üzerinde doğruluğu ve sınırları denenmiştir. Ayrıca orjinal işaret ve hücresel ağın bulduğu bileşenlerden oluşan işaret arasındaki farkları ve optimizasyonun derecesini ayrıntılı olarak görmek amacıyla ANALİZ adlı bir program da hazırlanmıştır. Bu bölümde işaretin Gauss bileşenlerine ayrılması, örnekleme periyoduna bağlı olarak Gauss parametrelerinin belirlenmesi, matematiksel yaklaşımlar ve önerilen devre mimarisi ayrıntılı olarak ele alınmıştır. Son olarak da Hopfield çözümü ile hücresel ağ çözümü karşılaştırılarak avantajlar ve dezavantajlar ortaya konulmuştur.

BÖLÜM II

NÖRAL AĞLARIN GELİŞİMİ

Tarih boyunca insanın doğada ve kendi yapısında gördüğü özellikleri anlamaya ve taklit etmeye çalışması, teknolojiyi doğurmuştur. Nöral ağlarla uğraşanlar da nörobiyoloji ve psikoloji konusunda gözlenen özellikleri bilimsel temellere dayanarak açıklamaya çalışmaktadırlar. İnsan davranışlarının kökenini ve beynin yapısını anlamak, tarihin ilk çağlarından beri bilim adamlarının yoğun çalışma konularından biri olmuştur. Son yıllarda yarı-iletken ve bilgisayar teknolojilerinde meydana gelen başdöndürücü gelişmeler, bilim adamlarına fikirlerini teoriden pratiğe geçirme fırsatı vermiş ve beynin yapısı ve çalışması hakkında geniş bilgiler edinmenin yanı sıra, temeli beyin organizasyonuna dayanan yeni bir devre teorisinin de gelişmesini sağlamıştır.

Teknolojide temel yaklaşım, önce bir mekanizmanın nasıl çalıştığını anlamak, daha sonra da aynı işi yapan bir mekanizma ortaya çıkarmaktır. İnsan gibi düşünen, çalışan ve hareket eden bir makina yapma isteği, yüzyıllardır bilim adamlarını meşgul etmektedir. Günümüz bilgisayarları da aynı yönelişin bir sonucu olarak, insanın "kendisine söylenen bir işi yapma" yeteneği taklit edilerek yaratılmışlardır. İnsanın beyinsel yeteneğinin en sınırlı olduğu matematiksel ve algoritmik hesaplama alanında çalışan bilgisayarlar, hesaplama işlemlerinde hız ve doğruluk açısından insanlardan yüzlerce kat daha başarılıdırlar. Günümüz teknolojisi ile neredeyse insan beynine yakın karmaşıklıkta bilgisayarlar yapılabilirse de, bu karmaşık sistemde hangi işlemlerin hangi yöntemlerle yapılıp, sistemin nasıl yönetileceği bilinmemektedir.

İnsan beyninin çok çeşitli işlevleri vardır. Tanıma, öğrenme, çıkarsama yapma, hesaplama olarak adlandırabileceğimiz bu işlevlerin çoğu günümüz bilgisayarları tarafından sağlanamaz. Bilgisayarlarımız değil insan bazı basit canlılar tarafından bile kolayca gerçekleştirilebilen fonksiyonlarda başarısız kalmakta veya sınırlı yapıdaki problemleri ancak çok büyük zaman ve kaynak harcayarak çözebilmektedirler. Örneğin insanların ve bazı hayvanların görüntü algılama birimleri, çok değişik ışık, gölge, bozulma, renk, örtülme vs. koşulları altında dahi cisimleri tanıyabilmektedir. Aynı şekilde duyma birimleri de sözcükleri veya konuşanın kimliğini ortam gürültüleri, tonlama, ses şiddeti vs gibi farklılıklardan bağımsız olarak tanıyabilmektedirler. 20.yy'da elimizde olan en iyi bilgisayarlarla dahi bu derece esnek tanıma algoritmaları geliştirmek mümkün değildir.

Beynin yapı taşları, nöron denilen hücreciklerdir. İnsanlarda ve hayvanlarda düşünsel fonksiyonların derecesi, beyinde ve sinir sisteminde bulunan nöron sayısı ile doğru orantılıdır. Nöronlar günümüz yarı-iletken elektronik devrelerin yayılım gecikmelerinden 10^5 - 10^6 oranında daha yavaş oldukları halde, en hızlı bilgisayarlardan daha hızlı işlem yapabilmektedirler. Hızlar arasındaki bu büyük fark, nöral devrelerin yapısındaki yoğun paralellikten kaynaklanmaktadır. Klasik bilgisayarların gelişimi sırasında ise daha farklı bir yöntem izlenmiştir. Von Neuman'ın mimarisinden başlayarak bilgisayarlarda bir tek işleme elemanı kullanılmış ve zamanla bu işleme elemanının çalışması hızlandırılarak daha hızlı bilgisayarlar elde edilmiştir. Son yıllarda paralel mimarilerin performansı arttırma yönünde gösterdiği ilerlemelere dayanılarak birden fazla işleme elemanından oluşan bilgisayarlar geliştirilmiş fakat işleyiciler birkaç tane olduğu anda bile sistemi yönetmek son derece zorlaşmıştır.

Nöronların elektronik devrelere olan diğer bir üstünlüğü de, kendi adaptif yapılarından kaynaklanan, giriş verilerindeki bozulmaları düzeltebilme yetenekleridir. Elektronik devreler ise, zaman zaman girişlerine gelen bazı gürültü seviyesindeki bozulmalarda bile hatalı sonuçlara yönelmektedirler.

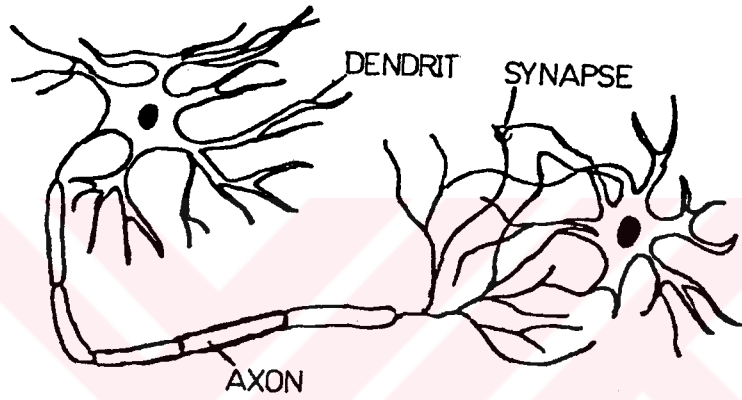
Nöral devrelerin en ilginç ve önemli özelliği ise deneyimle öğrenebilme özellikleridir. Daha önce öğrendikleri bir bilgiyi eksik veya bozuk bir giriş verildiği zaman bile yeniden üretebilir veya giriş kümesinin karakteristik özelliklerini çıkartarak aynı kümeden fakat daha önce görmedikleri bilgileri de doğru olarak ortaya çıkarabilirler. Diğer bir üstünlükleri ise lineer-olmayan bir karakteristiğe sahip olmaları nedeniyle gerçek dünya problemlerine optimal lineer tekniklerden çok daha doğru çözümler getirebilmeleridir. Nöronların sahip olduğu analog yapı, gerçek dünya problemlerine bilgisayarların sayısal yapısından daha fazla uyum göstermektedir.

2.1 Biyolojik Bulgular

Özel devre yapılarına girmeden önce, gerçek nöron yapısı ve beyin organizasyonu konusunda bazı bilgiler vermek faydalı olacaktır. Beyin temel fonksiyonların yerleşimi açısından sağ ve sol yarımkürelere ayrılır. Sol yarımküre daha çok sıralı ve kısa süreli çıkarsamalarda aktifleşirken, sağ yarımküre ise tanıma ve parçalardan bütünü elde etme türündeki düşünsel faaliyetlerde bulunmaktadır[1]. Yarımküreler de kendi içlerinde herbiri görme, konuşma, hareket gibi ayrı işlemleri denetleyen alt kümelerden oluşmuşlardır.

Canlılarda yaşamsal ve düşünsel faaliyetleri yürüten en küçük birime nöron adı verilir. Bir canlıda bulunan nöron sayısı yaklaşık olarak o canlının düşünsel yeteneğini de belirler. İnsanda yaklaşık 10^{10} adet

nöron olduğu tahmin edilmektedir. Nöronlar sadece beyinde olmayıp merkezi sinir sistemi üzerinde tüm vücuda yayılmışlardır. Bir nöronun giriş birimlerine dendrit, çıkış birimlerine ise akson adı verilir. Bir nöronun çıkışından diğer bir nöronun girişine giden bağlantıya ise snaps adı verilir. Nöronlar arasında bilgi transferi, günümüzde hala tam olarak açıklanamayan



Şekil 2.1 Nöron

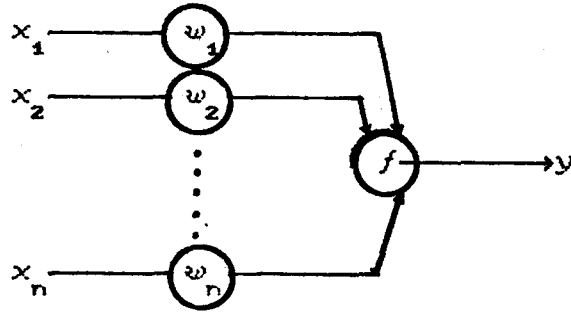
zar yapılar ve biyokimyasal tepkimeler yoluyla yapılmaktadır. Tepkimelerin ve nöronun giriş/çıkış fonksiyonlarının, deneysel ve çıkarımsal bilgilere dayanılarak modellenmesiyle pek çok yapay nöron modeli elde edilmiştir. En genel haliyle nöron, girişlerine gelen snaptik ağırlıklı potansiyel toplamın lineer-olmayan bir fonksiyonunu çıkışına yansıtan aktif bir eleman olarak modellenenebilir (Şekil 2.2).

Nöral devrelerde öğrenme 2'ye ayrılmaktadır :

1) Kısa zamanlı öğrenme: Snaptik bağlantılardaki ağırlıkların değişmesi;

2) Uzun zamanlı öğrenme: Snaptik bağlantı yollarının topolojisinin değişmesi.

Merkezi sinir sisteminde her nöronun çok sayıda çevre



Şekil 2.2 Temel Nöron Modeli

nörondan aldığı girişlere dayanarak çalıştığı deneylerle belirlenmiştir. Örneğin beyinde temel gruplardan biri olan piramit grubunda, her nöronun ortalama 2000-10,000 girişi vardır. Anatomik çalışmalar her nöronun en az 1000 komşu nörona bağlı olduğunu ve bu nöron kümelerinin de uzun aksonlar yardımıyla birbirleriyle karşılıklı etkileşimde bulunduklarını kanıtlamıştır[2]. İçerdiği nöron sayısı tam olarak bilinmeyen bu nöron kümelerinden beyinde çok sayıda bulunduğu sanılmaktadır. Merkezi sinir sistemi, son derece karmaşık bölgesel ve genel kimyasal etkileşimlerden oluşmuş, yoğun bir geribesleme sistemine sahiptir. Öteyandan çeşitli elektrofiziksel bulgular da bazı nöronların, belirli olaylara ve algılara doğrudan cevap verebilen bireysel işlemciler olduklarını göstermektedirler.

Biyolojide çok çeşitli nöron tipleri tanımlanmıştır. Sadece yüksek sınıftan hayvanlar örnek olarak alınırsa, nöron için şu tanımlı verebiliriz : Nöron, uygun giriş değerlerine karşı değişken frekanslı darbe sekansları üreten bir kontrollü darbe osilatörü olarak tanımlanabilir. Nöronların hangi fonksiyonel kurallara göre çalıştığı, bilinen teorilere ve deney sonuçlarına göre kısmen de olsa açıklanabilmektedir. Öncelikle her nöronun çıkışında oluşacak darbe katarının frekansı bir alt ve üst limite sahiptir. Ortalama osilasyon frekansı ise nöronun toplam potansiyelinin monoton bir fonksiyonudur.

Nöron girişindeki snaptik bağlantılar ise 2'ye ayrılır: canlandırıcı ve susturucu bağlantılar. Canlandırıcı bağlantılar nöronun çıkışında bir darbe katarı oluştururken susturucu girişler nöron çıkışındaki darbe üretimini durdurur. Susturucu girişlerin gücü, çoğu zaman canlandırıcı girişlerin birkaç katı mertebesinde-dir. Susturucu girişlerin nöron çıkışına etkisi tamamen lineer-olmayan yapıdadır ve toplama işlemiyle ifade edilemez.

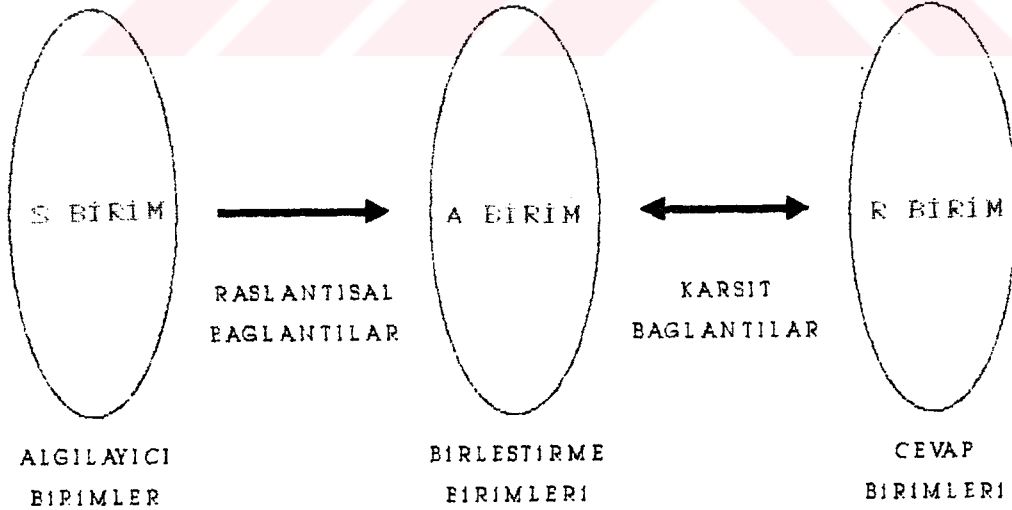
Yukarıda belirtildiği gibi nöron dinamiğinin analizi, aktif zarlara ve bunların biyokimyasal analizine dayanmaktadır. Çalışmaların yönelişini belirleyen diğer önemli bir nokta da, nöronların karşılıklı ilişkilerini belirleyen fonksiyonel kuralların, izole edilmiş tekil nöronlarla ölçülemeyeceğidir. Gerçek değerler ve yapı ancak nöron yaşayan bir organizmanın parçası iken elde edilir.

2.2 Tarihçe

Nöral ağlar konusunda literatüre geçmiş ilk büyük çalışma 1943 yılında McCulloch&Pitts tarafından yapılmıştır[3]. McCulloch&Pitts nöronları sabit eşik değerli lojik elemanlar olarak modellemişlerdi. Nöronların canlandırıcı bağlantıları eşit ağırlıklara sahip snapslar, susturucu bağlantıları ise nöronu girişlerindeki değerlerden bağımsız olarak pasif konuma çeken girişlerdi. Tüm devre ortalama bir nöron gecikmesi periyoduyla senkronize edilirdi. Her periyotta nöronlar girişlerinin ağırlıklı toplamını alır, susturucu giriş pasif konumda ise ağırlıklı toplamla eşik değerini karşılaştırırdı. Toplam eşikten fazlaysa nöron bir sonraki periyotta aktif konuma geçerdi. Bu çalışmanın en büyük önemi, tek başına çok basit bir yapıya sahip olan elemanların bir araya gelmesiyle son derece karmaşık lojik fonksiyonların gerçekleştirilebileceğini göstermesiydi. Gerçekte nöronlar,

McCullogh&Pitts'in stokastik lojik modelinden çok farklı bir yapıya sahiptirler. 1947 yılında gene aynı ikili diğer bir makalelerinde [4], ilkinden farklı olarak "Collucular Computer" yaklaşımını kullanmışlardır. Algılayıcı birimlerden gelen verilerle uzaysal bir harita oluşturan ve bu haritanın ağırlık merkezini hesaplayarak algılayıcıyı bu merkeze yöneltten bu yaklaşım, bugün de modern nörolojinin algılamaya getirdiği açıklamalarla aynı doğrultudadır.

McCullogh&Pitts'in bu iki makalesi araştırmacılar arasında büyük ilgi uyandırdı. Nöral ağlar konusunda ikinci büyük çıkışı ise 1958 yılında Perceptron modeliyle Rosenblatt yarattı[5]. Perceptron, karmaşık adaptif davranışlarda bulunma yeteneği olan bir öğrenme makinasıydı. En basit Perceptron 3 katmandan oluşmaktadır (Şekil 2.3).



Şekil 2.3 Perceptron Modeli

Perceptronda R-Birim'deki her bir elemanın, A-Birim'deki aktif elemanların ağırlıklı snaptik çarpımlarından oluşan N adet girişi vardır ve bu girişler N.dereceden bir hiperküp oluşturmaktadır. Girişlerin ağırlıklı toplamaları

Perceptronun bu başarısızlığından sonra nöral ağlar üzerine yapılan çalışmalar birkaç bilimadaminin kişisel gayreti ile son derece dağınık ve yavaş olarak 1967'den 1982'ye kadar ilerledi. 1980'li yılların başında Hopfield adlı bir nörobiyologun yayınladığı üç makale [7,8,9] nöral ağ çalışmalarını tekrar canlandırdı.

Nöral ağların gelişiminde önemli kilometre taşlarından biri de Rus matematikçisi Kolmogorov'un ispatını yaptığı teoremdir. 1957 yılında yayınlanan bu teorem ilk anda matematik dünyasında büyük şok yaratmasına rağmen, genel matematik alanında başka teoremlerin tanıtlanmasında kullanılan bir yardımcı teorem olmaktan öteye gidememiştir. Nörobilgisayarlar alanında ise Kolmogorov Teoremi, nöral yapılar ile herhangi bir fonksiyonun gerçekleştirilebileceğini gösterdiği için çok önemlidir. Teorem dönüşüm ağlarının varlığını ispatlamaktadır. Dönüşüm nöral ağları, bir fonksiyonu/dönüşümü gerçeklemek veya yaklaşığını bulmak amacıyla kendi organizasyonunu değiştiren devrelerdir.

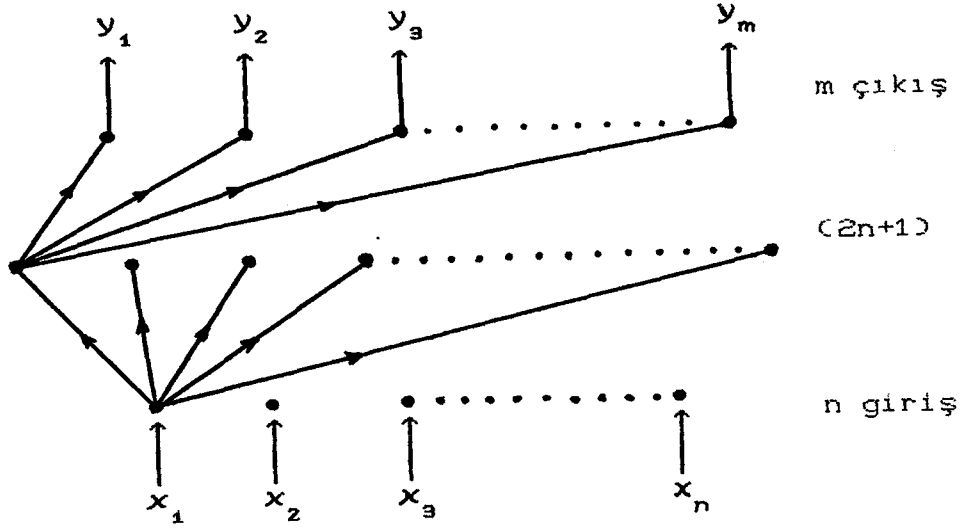
Kolmogorov'un Dönüştürücü Nöral Devreler Varlık Teoremi:

Verilen herhangi bir $f: [0,1]^n \rightarrow \mathbb{R}^m$, $f(x)=y$ fonksiyonu, ilk katında n giriş elemanı (x girişleri), ikinci katında $(2n+1)$ elemanı ve üçüncü katında ise m çıkış elemanı (y çıkışları) olan üç katmanlı bir nöral ağ ile tam olarak gerçekleştirilebilir.

Teoremin tanıtı [10,11] numaralı referanslarda bulunabilir. Kolmogorov'un teoreminden yararlanarak oluşturulan nöral ağ mimarisi Şekil 2.5'de görülmektedir. İkinci katmandaki gizli elemanların transfer fonksiyonu

$$z_k = \sum_{j=1}^N \lambda^k \phi(x_j + k\varepsilon) + k \quad (2.1)$$

Transfer fonksiyonundaki reel sayı λ ve monoton artan reel, sürekli fonksiyon ϕ , f fonksiyonundan bağımsız



Şekil 2.5 Kolmogorov Devresinin Yapısı

olarak belirlenir. δ rastgele seçilen pozitif bir sabit ve ε da $0 < \varepsilon < \delta$ değerleri arasında bir rasyonel sayıdır.

Üçüncü katmandaki çıkış elemanlarının transfer fonksiyonu ise

$$y_i = \sum_{k=1}^{2n+1} g_i(z_k) \quad (2.2)$$

şeklindedir. g_i fonksiyonları sürekli, reel ve f ve ε 'a bağlı fonksiyonlardır. Teoremle birlikte g_i fonksiyonlarının ve ε 'nın hesaplanmasına ait herhangi bir örnek veya çözüm metodu verilmediğinden, Kolmogorov teoremi ortaya atıldığından bu yana sadece bir varlık teoremi olarak kalmıştır. Teorik olarak böyle bir devrenin varlığı tanıtılmakta fakat nasıl hesaplanacağı bildirilmemektedir.

Nöral ağlar konusunda McCulloch&Pitts, Rosenblatt ve Kolmogorov'un ardından pek çok bilim adamı başlarda yavaş ve dağınık olsa da çalışmalar yapmışlar ve yıllar geçtikçe nörobilgisayarlar alanında önemli gelişmeler

Günümüzde gittikçe yoğunlaşan nöral ağ çalışmaları, var olan modelleri geliştirmek yanında pek çok yeni devre modelini de ortaya çıkarmaktadır. En genel haliyle nöral ağları üç büyük sınıfa ayırabiliriz [13].

İleribeslemeli Ağlar: Giriş işaretlerini dönüştürerek çıkış işaretlerini elde eden ağlardır. Dönüşüm şekli, ağ parametrelerinin dışarıdan ayarlanması ile belirlenir.

Geribeslemeli Ağlar: Giriş işaretleri devrenin başlangıç aktivitesini belirler ve durum geçişleri yatıştıktan sonra ağ, asimptotik olarak son duruma yakınsar.

Yarışan Eğitimsiz Öz-Örgütlemeli Ağlar: Komşu nöronlar, aktiviteleri ile karşılıklı etkileşimde bulunarak farklı giriş işaretlerine göre kendi kendilerini uyarırlar.

Genellikle ilk iki grupta yer alan nöral ağların öğrenme yetenekleri yoktur, giriş işaretleri üzerinde lineer ve lineer olmayan dönüşümleri gerçekleştirirler. Üçüncü gruptakiler ise belirli bir öğrenme kuralına bağlı olarak, giriş verilerine göre kendi yapılarını değiştirirler. Öğrenme de iki türde yapılabilir:

Eğitilmiş öğrenme: Ağa, giriş bilgisi ve bu girişe karşın üretmesini istediğimiz çıkış bilgisi veya giriş ve çıkışa bağlı bir eğitim işareti verilerek devre adım adım eğitilir.

Eğitimsiz öğrenme: Ağa sadece giriş bilgileri verilerek, bu bilgi grubuna ait özelliklere göre ağın organizasyonunu kendi kendine değiştirmesi sağlanır.

Öğrenme işleminin, sinaptik ağırlık vektörlerinin belirli bir kurala göre değiştirilmesine dayandığı belirtilmişti. Bu genel öğrenme kuralını şu şekilde

Günümüzde gittikçe yoğunlaşan nöral ağ çalışmaları, var olan modelleri geliştirmek yanında pek çok yeni devre modelini de ortaya çıkarmaktadır. En genel haliyle nöral ağları üç büyük sınıfa ayırabiliriz [13].

İleribeslemeli Ağlar: Giriş işaretlerini dönüştürerek çıkış işaretlerini elde eden ağlardır. Dönüşüm şekli, ağ parametrelerinin dışarıdan ayarlanması ile belirlenir.

Geribeslemeli Ağlar: Giriş işaretleri devrenin başlangıç aktivitesini belirler ve durum geçişleri yatıştıktan sonra ağ, asimptotik olarak son duruma yakınsar.

Yarışan Eğitimsiz Öz-Örgütlemeli Ağlar: Komşu nöronlar, aktiviteleri ile karşılıklı etkileşimde bulunarak farklı giriş işaretlerine göre kendi kendilerini uyarırlar.

Genellikle ilk iki grupta yer alan nöral ağların öğrenme yetenekleri yoktur, giriş işaretleri üzerinde lineer ve lineer olmayan dönüşümleri gerçekleştirirler. Üçüncü gruptakiler ise belirli bir öğrenme kuralına bağlı olarak, giriş verilerine göre kendi yapılarını değiştirirler. Öğrenme de iki türde yapılabilir:

Eğitilmiş öğrenme: Ağa, giriş bilgisi ve bu girişe karşın üretmesini istediğimiz çıkış bilgisi veya giriş ve çıkışa bağlı bir eğitim işareti verilerek devre adım adım eğitilir.

Eğitimsiz öğrenme: Ağa sadece giriş bilgileri verilerek, bu bilgi grubuna ait özelliklere göre ağın organizasyonunu kendi kendine değiştirmesi sağlanır.

Öğrenme işleminin, sinaptik ağırlık vektörlerinin belirli bir kurala göre değiştirilmesine dayandığı belirtilmişti. Bu genel öğrenme kuralını şu şekilde

verebiliriz: Sinaptik ağırlık vektörü w , x giriş vektörü ve r eğitim işaretinin çarpımıyla orantılı olarak artar ve aynı zamanda da bir miktar aşınmaya uğrar [14].

Sinaptik ağırlıklardaki değişimin sürekli zamandaki ifadesi,

$$w(t) = -\epsilon \cdot w(t) + r \cdot x(t) \quad (2.3)$$

şeklinde ve bu denkleme **Öğrenme denklemleri** denir. Eğitim işaretinin seçimine göre literatürde farklı öğrenme kuralları ortaya atılmıştır. Bunlardan bazıları aşağıda yer almaktadır.

y : nöron çıkışı

h : eşik değeri

d : istenen nöron çıkışı

olmak üzere;

Hebb Öğrenme Kuralı: $r = y \cdot f(wx - h)$

Perceptron Öğrenme Kuralı: $r = d - y$

Korelasyon Öğrenme Kuralı: $r = d$

Widrow Öğrenme Kuralı: $r = wx - d$ (En küçük Kareler)

Nörobiyolojinin bugün ulaştığı noktada, insan bey-
ninde farklı türden nöronların bulunduğu ve her nöronun
yakın çevresindeki elemanlarla, belirli ve o nöron türü
için sabit bir ilişki içinde olduğu bilinmektedir. Bu
kurallara mikroskobik bağlantı kuralı denir. Nöron sınıf-
ları arasında da buna benzer bir bağlantı vardır ve sı-
nıflar arası yapıya da makroskobik bağlantı kuralı adı
verilir. Bağlantıların gücü, türden türe farklılık
gösterir ve öğrenmeyle değişir [15]. k adet nöron türün-
den oluşmuş bir nöron kümesinde aşağıdaki gibi bir yapı
vardır :

$$TX + X = W \cdot f(X) + b \quad (2.4)$$

Burada $X = [X_1 \ X_2 \ \dots \ X_k]^T$

$$= [x_1 \ x_2 \ \dots \ x_{n1}, x_{n1+1} \ x_{n1+2} \ \dots \ x_{n1+n2},$$
$$x_{N-nk-1} \ x_{N-nk-2} \ \dots \ x_N]$$

şeklinde tanımlanan nöronların zar potansiyellerini gösteren vektördür. Toplam nöron sayısı

$$N = \sum_{i=1}^k n_i \quad (2.5)$$

olarak verilir. Devrenin bağlantı matrisi W ise

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1k} \\ w_{21} & w_{22} & \dots & w_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k1} & w_{k2} & \dots & w_{kk} \end{bmatrix}$$

şeklinde gruplar arası bağlantı matrislerinden oluşmuştur. (2.4)'deki diğer matris

$$T = \text{diag} [\tau_i] \quad i = 1, 2, \dots, N$$

pozitif bir matristir ve zar zaman sabitlerini içerir. b ise sabit bir vektördür. $f(X)$ nöronsal fonksiyonunun tanımını ise şöyle verebiliriz :

Tanım 2.1: $f: \mathbb{R}^N \rightarrow \mathbb{R}_+^N$ olmak üzere sürekli, monoton artan Lipschitz koşulunu sağlayan ve $f(0)=0$ olmak üzere bir θ noktasını içeren fonksiyonlar sınıfına nöronsal fonksiyonlar denir.

f fonksiyonunun Lipschitz koşulunu sağlaması, diferansiyel denklemlerin çözümünün tekliğini garanti etmektedir.

2.3 Nöromorfik Elektronik Sistemler

Elektroniğin son 50 yıldaki gelişimine bakılırsa, bir hesaplama yönteminin maliyetinin, doğrudan bu yöntem için harcanan enerjiyle orantılı olduğu görülür. Teknolojinin yönelişi de buna uygun olarak, bir devrede harcanan enerjiyi azaltma doğrultusundadır. Vakum lambalardan transistöre, entegre devrelere, NMOS teknolojidenden CMOS teknolojiye ve günümüzün mikronaltı teknolojisine geçiş, bu amacın ürünüdür. Oysa günümüzün en hızlı elektronik devreleri bile insan beyninin hızı ve performansı karşısında yetersiz kalmaktadır.

Günümüz elektronik devreleri ile sinir sistemi arasındaki en büyük fark, organizasyonlarında görülür. Bugün tipik bir mikroişlemci, bir işlemi yaklaşık 10^{-7} sn de yapmakta ve 1 W enerji harcamaktadır, dolayısıyla bir hesaplama için 1 J enerji harcadığını söyleyebiliriz. Gelecek yıllarda yarı-iletken teknolojisinin 0.1μ 'a kadar düşeceğini öngörebiliriz. 0.1μ 'dan daha küçük boyutlarda transistör yapmak fiziksel sınırları zorlar ve elemanın çalışmasını engeller. Bu durumda en iyimser tahminle elektronik devreler, tek bir hesaplamayı yapmak için entegre devre üzerinde 10^{-9} J, tüm bilgisayarda ise 10^{-7} J maliyet getirecektir. İnsan beyninin hesaplama yeteneği ile bu değerleri karşılaştırdığımızda ise insan beyninin günümüz sayısal tekniklerinden 1 milyar kat, tahmin edebileceğimiz en hızlı teknolojiden de 10 milyon kat daha efektif olduğu ortaya çıkar.

Bireysel olarak her iki sistemin de yapıtaşlarını incelediğimizde, nöronlar ve transistörler arasında, performans açısından çok büyük farklar olmadığı görülür. Sinir sistemi ile elektronik devreler arasındaki en büyük fark, yapıtaşlarının organizasyonundan gelmektedir. Sinir sisteminde bir hesaplama yapmak için ancak yakın çevredeki elemanlar kullanılır. Elektronik devrelerde ise o devreyi oluşturan tüm transistörler hesaplama

işlemine fiziksel olarak katılırlar. Veriyi tüm devre yerine sadece yakın çevreye göndererek çalışacak şekilde tasarlanmış sayısal devrelerde güç tüketiminin 10-100 kat azaldığı görülmüştür[18].

Sinir sistemi ile elektrik devreleri arasındaki diğer bir fark da verilerin gösteriminden gelir. Sinir sistemi, zar potansiyellerinin analog ve sürekli olması nedeniyle hastalıklı verilere karşı toleranslı bir yapıya sahiptir. Oysa sayısal elektronikte tüm veriler 0 ve 1 kalıplarına uymak zorundadır. Gerçek dünyadaki tüm verileri, gürültüleri, hastalıklı girişleri sadece bu iki sayı ile ifade etmeye kalktığımızda toleranstan, hataları düzeltmekten söz etme hakkımız da ortadan kalkmaktadır. Veri gösteriminin yanı sıra, bu verilerle yapılacak ilkel işlemlerin de belirlenmesi gerekir. Sayısal devrelerde en ilkel fonksiyonlar AND, OR ve NOT fonksiyonlarıdır. Nöral yapılarda ise çok daha farklı, çok daha ilginç temel fonksiyonlar vardır. Exponansiyel artan gerilim üretmek, zamana göre integral almak ve sıfır maliyetli toplama yapmak gibi bizim alışık olduğumuz ilkellerden çok farklı olan bu ilkelleri kendi bilinen teorilerimize çevirmek yerine, bunlara uygun yeni gösterimler bulmak elektronik teknolojisinde çığır açacaktır. İşte analog devre elemanlarından oluşmuş fakat sinir sisteminin organizasyonuna sahip bu gibi sistemlere nöromorfik sistemler denir.

Nöromorfik sistemler son yıllarda gerek ayırık elemanlarla gerekse silikon üzerinde oluşturulmaya çalışılmıştır. Ayırık elemanlarla 20'den fazla nöron içeren devreler yapmak, hızla artan snaptik bağlantılar nedeniyle pek olurlu değildir. CMOS teknoloji ile yapılan nöral devrelerle ise kontrol sistemleri, sekans üreteçleri, parlak ışıkları takip eden retina yongaları, üzerlerindeki görüntüye odaklanan retina yongaları, hareket sezen, görüntü geliştiren devreler yapılmıştır. Birkaç eksenli nöral devrelerle çeşitli düzeylerde çözünürlüğe sahip, derinliği sezebilen yongalar yapılmıştır.

Sesin özelliklerini yarıştıran, sesin yatay ve düşey bileşenlerini ayıran, görüntüyü sese çeviren yongaları da bu gruba sokabiliriz. Tüm bu yongaların performansını ölçtiğimizde gene de beyinin performansından 10^5 kat daha yavaş olduklarını görürüz.

Performansı arttırmanın bir yolu, yongaları kılıflara koyup baskılı devre üzerinde birleştirmek yerine devreyi silikon üzerinde kurmaktır (Wafer-Scale Integration). Bir wafer üzerinde bugün yaklaşık 10^8 elemandan oluşan birkaç yüz devre sığdırılabilmektedir. Sayısal devrelerin wafer üzerinde kurulmasını engelleyen iki büyük etmen vardır :

1) Devrede tek bir transistör dahi bozulsa devre çalışamaz hale gelir.

2) Bu yoğunlukta bir sayısal devrede yaklaşık 100W'lık bir enerji harcanacağından soğutma çok büyük problem olmaktadır.

Oysa analog nöral devreler yapıları nedeniyle bu iki olumsuz noktayı ortadan kaldırırlar. Öncelikle nöral ağlar paralel çalışan ve öğrenebilen elemanlardan oluştuğu için birkaç yüz nöron işlevini yitirse bile devre çalışmaya devam edecek ve bu nöronları kullanmamayı öğrenecektir. İkinci olarak, analog nöral ağların güç tüketimi sayısal devrelere oranla çok düşük olduğundan normal hava akımı, oluşan ısıyı dağıtmaya yeterli olacaktır.

Silikon üzerinde yapılan devreler, 3-boyutlu bir yapıyı 2-boyutlu bir yapıya çevirmekten doğan birtakım sorunları da beraberlerinde getirirler. En önemli problem, wafer üzerinde yolların kapladığı alan ile nöronların kapladığı alanların dengesidir. Eğer tüm yolların devreyi oluşturan alan kadar uzun olabileceğini düşünersek, nöron sayısının artmasıyla yollar yapının büyük bir bölümünü kaplayacaktır. Beyinde yoğunlukla bölgesel bağlantıların kullanılmasıyla bu problem ortadan kalkmıştır. Aksi takdirde beyin büyük bir bölümünü yollar kaplayacak ve nöronlara yer kalmayacaktır.

Gerçekten de korteks düzleştirilirse 1m uzunluğunda ve 1mm kalınlığında bir hacim oluşturur. 1mm kalınlığın yaklaşık yarısını yollar (beyaz madde), yarısını da hesaplama elemanları (gri madde) oluşturmaktadır. Temel olarak 2-boyutlu stratejinin de 3-boyutlu strateji kadar efektif olabileceğini söyleyebiliriz. Gelecekte iki boyutlu waferlar üzerinde analog, adaptif nöral devreler gerçekleştirilerek beyin performansına çok daha yakın sistemler yapılması olasılığı çok büyüktür.



BÖLÜM III

HÜCRESEL NÖRAL AĞLARIN TANIMI

Hücresel nöral ağ teorisi, 1988 yılında Leon Chua ve Lin Yang tarafından ortaya atıldı [17,18]. Kısa zamanda teorisyenlerin ilgisini çeken bu yeni nöral ağ mimarisi, son yıllarda hızla gelişmeye ve geliştikçe de umut verici sonuçlar elde edilmeye başlanmıştır.

Hücresel nöral ağlar literatürde, matematiksel teorisi en sağlam temellere dayanan nöral ağ modellerinden biridir. Lineer-olmayan analog dizi olarak da adlandırılan bu yapının en önemli özelliği, nöronlar arası bağlantıların insan doğasına uygun olarak bölgesel yapıda olmasıdır. Hücresel ağlarda her nöron, lineer direnç, kapasite ve gerilim kontrollü akım kaynaklarından oluşur. Nöronlar iki veya daha fazla boyutlu uzaya düzgün olarak sıralanırlar. Her hücre sadece kendisine en yakın hücrelerle snaptik bağlantı içindedir. Bütün nöronlar birbirlerine direkt olarak bağlanmasalar da, geribeslemeli devrenin son duruma yakınsaması sırasında bütün hücreler birbirini dolaylı yoldan etkilerler.

Hücresel ağların programlanması, küçük komşuluk gruplarının snaptik bağlantı ağırlıklarının belirlenmesidir. Komşuluk grubuna ait bağlantı ağırlığı (Örneğin: 3×3 matris-1 komşuluk) belirlendikten sonra gerçek devrenin boyutundan bağımsız olarak, programlanan fonksiyon gerçekleştirilir. Hücresel ağların en dikkat çekici özelliği, bu en yakın komşu bağlantı yapısındadır. Sinir sistemimizde de her nöronun yaklaşık 1000 komşu nöronla bağlantılı olduğu bilinmektedir. Hücreler arası bağlantıların komşuluk düzeninde olması, hücresel ağların VLSI

gerçeklemelerine de büyük kolaylık getirmiştir. Gittide gelişen yarı-iletken teknolojileriyle devre elemanlarının silikon üzerinde kapladıkları yerler sürekli küçültülmektedir. Oysa bağlantı yollarının küçültülmesine imkan yoktur ve eğer tam bağlı nöral ağ yapıları kullanılmaya devam edilirse, ağılardaki nöron sayısını kısıtlayan en önemli faktör yarı-iletken teknolojisi değil, yolların kapladığı yer olacaktır.

Hücresel nöral ağların ilk uygulama alanı görüntü işleme olmuştur. Büyük Japon firmaları ve bazı Amerikan araştırma kurumlarının desteğiyle başlayan çalışmaların gerçek amacı, el yazısı Japon karakterlerini tanıyan bir sistem geliştirmektir. Bilindiği gibi Japon alfabesinde 3000'den fazla harf vardır. Harflerin elle yazılması sebebiyle giriş verilerinde meydana gelecek bozulmaları düzeltebilecek ve çok sayıda harfin içinden gerçek harfi seçebilecek bir sistem geliştirmek, sayısal bilgisayarlar ile oldukça zor bir işti. Hücresel ağlarda ise, görüntü üzerindeki her piksel devrede bir nörona karşı düşer. Çeşitli programlarla giriş görüntüsünün özellikleri ayrıştırılır ve bu özellikler başka bir tanıma algoritmasına verilerek doğru harf seçilir.

Hücresel nöral ağların genel çalışma prensibi, girişlerine verilen görüntünün lineer-olmayan bir dönüşümünü çıkışılarına vermesidir. Bağlantı ağırlıkları sabittir yani tek başlarına öğrenme yetenekleri yoktur. Dönüşümün son duruma yakınsama süresi devrenin boyutundan bağımsızdır ve mikrosaniyeler mertebesinde. Nöron yapısı açısından bir karşılaştırma yapılırsa, hücre-sel ağlara en yakın nöral ağ modeli Hopfield devreleridir. İki devre mimarisi arasındaki en önemli farklar snaptik bağlantıların bölgesel yapısı ve nöronun iç yapısıdır. Ayrıca hücre-sel ağlar Hopfield devrelerinden daha kararlı ve yakınsak bir yapıya sahiptirler.

Günümüzde hücre-sel ağlar görüntü işlemenin yanında hareket sezme, asosiyatif bellek, optimizasyon

problemleri gibi çeşitli konulara uygulanmaktadır.

3.1 Hücre Yapısı Ve Devre Tanımı

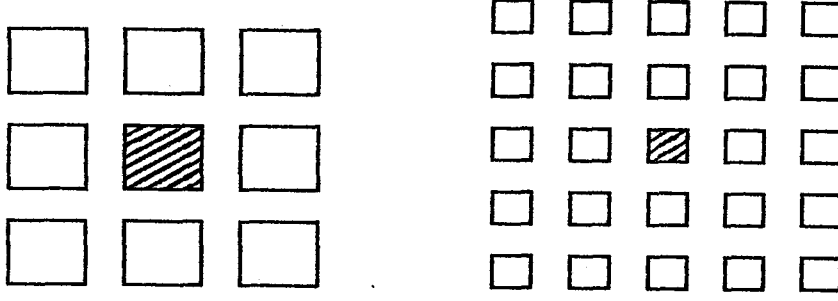
Hücreesel nöral ağların en küçük birimine hücre (cell) adı verilir. Hücreler, devrenin boyutuna göre iki boyutlu uzayda düzgün olarak yerleştirilmişlerdir. Her hücre lineer devre elemanlarından ve bir adet lineer olmayan gerilim kontrollü akım kaynağından oluşmuştur. Hücreesel ağlar yapısı nedeniyle "cellular automata" teorisine benzetilebilir. İki teori arasındaki en önemli fark ise, cellular automata teorisinin ayrık zamanda, hücreesel ağların ise sürekli zamanda tanımlanmış olmasıdır.

$M \times N$ boyutlu, M satır ve N sütundan oluşmuş bir devreyi gözönüne alalım. i . satır ve j . sütundaki hücreye (i, j) hücresi denilir ve $C(i, j)$ ile gösterilir. Hücrelerin birbirleriyle bağlantısı komşuluk düzenine göre belirlenmektedir.

Tanım 3.1: Bir hücreesel ağda $C(i, j)$ hücresinin r -komşuluğu (r -Neighborhood)

$$N_r(i, j) = \left[C(k, l) \mid \max \left\{ |k-i|, |l-j| \right\} \leq r \right]$$
$$\begin{aligned} 1 &\leq k \leq M \\ 1 &\leq l \leq N \\ r &\in \mathbb{N}^+ \end{aligned}$$

olarak tanımlanır. Şekil 3.1'de $C(i, j)$ hücresinin $r=1$ ve $r=2$ komşulukları gösterilmiştir. Genellikle 1-komşuluk yerine "3x3 komşuluk", 2-komşuluk yerine de "5x5 komşuluk" deyimleri kullanılır. r -komşuluk içinde $(2r+1) \times (2r+1)$ adet hücre bulunur. Komşuluk sistemi simetri özelliğine sahiptir yani $C(k, l) \in N_r(i, j)$ ise $C(i, j) \in N_r(k, l)$ denir.



Şekil 3.1 r=1 ve r=2 Komşulukları

Simetri Özelliği:

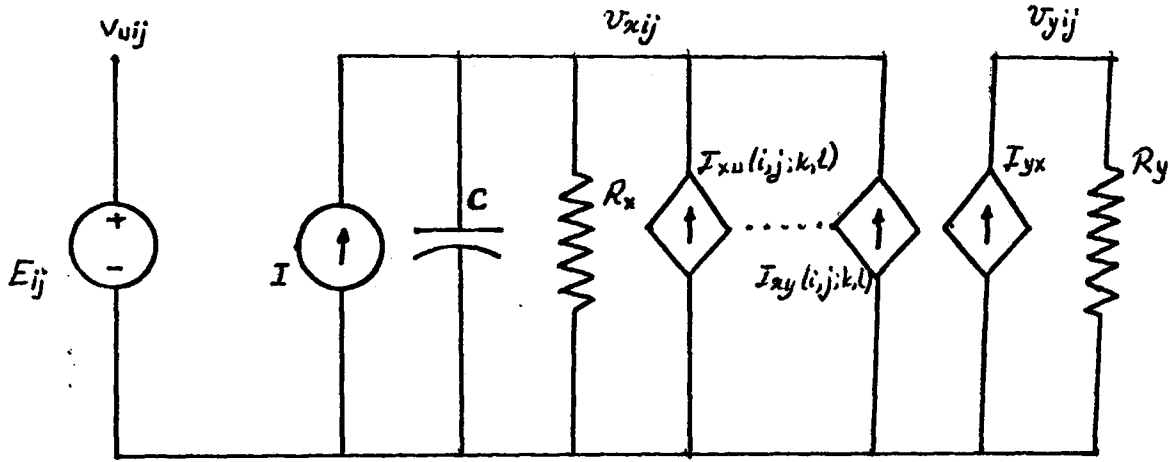
$$CC(i,j) \in N_r(k,l) \Leftrightarrow CC(k,l) \in N_r(i,j) \quad \forall CC(i,j), CC(k,l), r$$

Tanıt:

$$\begin{aligned} CC(i,j) \in N_r(k,l) &\Leftrightarrow \max \left\{ |i-k|, |j-l| \right\} \leq r \\ &\Leftrightarrow \max \left\{ |-(k-l)|, |-(l-j)| \right\} \leq r \\ &\Leftrightarrow \max \left\{ |k-l|, |l-j| \right\} \leq r \\ &\Leftrightarrow CC(k,l) \in N_r(i,j) \end{aligned}$$

Hücresel nöral ağ programlanırken devre için bir komşuluk sayısı da seçilir (Çoğunlukla $r=1$). Bu aşamadan sonra her hücre sadece kendi komşuluk grubunda olan hücrelerle doğrudan ilişkide bulunur ve diğer bir deyişle sadece komşuları ile snaptik bağlantıları vardır. Şimdi hücre yapısını inceleyelim.

Tanım 3.2: Hücre yapısı, Şekil 3.2'de ayrıntılı olarak görülmektedir. Hücrede sırasıyla giriş, durum ve çıkışa karşı düşen u, x ve y düğümleri bulunur. Hücrede bulunan elemanlar E_{ij} sabit gerilim kaynağı, I sabit akım kaynağı, C lineer kapasite, R_x ve R_y lineer dirençler, m komşu hücrelerin sayısına eşit olmak üzere en fazla $2m$ adet $I_{xu}(i,j;k,l)$ ve $I_{xy}(i,j;k,l)$ akım kaynağı, son olarak da I_{yx} lineer-olmayan gerilim kontrollü akım kaynağıdır. Komşu hücrelerle bağlantı, kontrol girişi v_{ukl}



Şekil 3.2 Hücre Yapısı

ve geribesleme gerilimi v_{ykl} 'nin ağırlıklı toplamaları üzerine kurulmuştur.

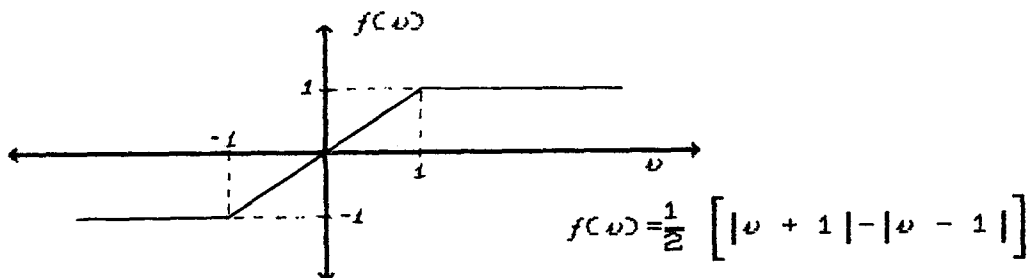
$$I_{xy}(i,j;k,l) = A(i,j;k,l)v_{ykl} \quad \forall C(k,l) \in N_r(i,j)$$

$$I_{xu}(i,j;k,l) = B(i,j;k,l)v_{ukl} \quad \forall C(k,l) \in N_r(i,j)$$

Hücredeki tek lineer-olmayan eleman ise I_{yx} parça-lineer gerilim kontrollü akım kaynağıdır.

$$I_{yx} = \frac{1}{R_y} \cdot f(v_{xij})$$

şeklinde ifade edilen kaynağın $f(\cdot)$ karakteristiği Şekil 3.3'de yer almaktadır.



Şekil 3.3 $f(\cdot)$ Karakteristiği

Hücreesel nöral ağların VLSI gerçeklemelerinde $f(\cdot)$ fonksiyonunun keskin karakteristiğini elde etmek mümkün

olmadığından, daha yumuşak ve sürekli bir fonksiyon olan sigmoid fonksiyonu kullanılır. Fonksiyonların değişimi teoriyi etkilemez.

Hücresel ağ dinamiğinin hem giriş kontrol, hem de çıkış geribesleme mekanizmaları vardır. Çıkış geribeslemesi $A(i,j;k,l)$ katsayısına, giriş kontrolü ise $B(i,j;k,l)$ katsayısına bağlıdır. Bu nedenle $A(i,j;k,l)$ katsayısına geribesleme operatörü, $B(i,j;k,l)$ katsayısına ise kontrol operatörü denir. Kullanılan mimari tarzına göre A ve B matrislerinin elemanları reel sayılar veya lineer-olmayan fonksiyonlar olabilirler.

Hücreler $M \times N$ boyutlu matris üzerindeki yerleşimlerine göre de isimlendirilirler. Devrenin kenar kısımlarında yer alan hücelere kenar hücreler, iç kısımlarında yer alanlara da iç hücreler adı verilir. Başka bir ifadeyle, $(2r+1)^2$ adet komşusu olan hücelere iç hücre denilir.

3.2 Küme Kalıbı ve Konvölüsyon Çarpımı

Hücresel ağların en büyük özelliğinin snaptik bağlantıların sadece en yakın komşularla kurulması olduğu belirtilmişti. Snaptik bağlantı ağırlıkları, geribesleme operatörü A ve kontrol operatörü B içinde bulunur. Ağırlıklar sadece bir komşuluk grubu içinde tanımlı olduğundan bu matrislere küme kalıbı denmektedir.

Tanım 3.3: Bir hücreyi komşularına bağlayan geribesleme ve kontrol operatörleri, hücrenin ağ içerisindeki yerleşiminden bağımsız olduklarından, katsayıları $A(i,j;k,l)$ gibi mutlak indisler yerine bağıl indislerle yazabiliriz.

$$A(i,j;k,l) = A(i-k,j-l)$$

şeklini alan ve bir matrisle ifade edilebilen bu A ve B katsayılarına küme kalıbı (cloning template) adı verilir. Hücrenin komşularıyla olan bağıl uzaklığına göre

matrisin içine yerleştirilmiş olan katsayılar, $r=1$ için aşağıdaki şekli alırlar :

$$A = \begin{bmatrix} AC(-1, -1) & AC(-1, 0) & AC(-1, 1) \\ AC(0, -1) & AC(0, 0) & AC(0, 1) \\ AC(1, -1) & AC(1, 0) & AC(1, 1) \end{bmatrix}$$

A matrisinin ortasındaki $AC(0, 0)$ elemanı hücrenin kendi üzerindeki geribeslemeyi, etrafındaki elemanlar ise uzayda yerleşimlerine göre komşu hücrelerle aralarındaki ağırlıkları göstermektedir. Bir devrenin programını, işlevini A ve B matrisleri ve I sabit akımı belirlediğinden, literatürde üçünün bileşimine de küme kalıbı denilmektedir.

Küme kalıbının tanımından sonra, bu matrisler üzerinde yapılacak işlemlerin de tanımlanması gerekir. Örneğin her hücrenin girişlerine, komşularının çıkışlarının ağırlıklı toplamı gelmektedir. Hücrenin x düğümündeki geribesleme akımı

$$\sum_{C(k,1) \in N_r(i,j)} AC(i,j;k,1) v_{yk}$$

ifadesinden hesaplanır. Esasta bu terim, küme kalıbı ile komşu hücrelerin çıkış gerilimlerinin çarpımından ibarettir. İfadeyi daha açık bir şekilde yazabilmek için konvolüsyon çarpımı tanımlanmıştır.

Tanım 3.4: Hücresel nöral ağın dinamik kuralını belirleyen herhangi bir T küme kalıbı için konvolüsyon çarpımı (*) şu şekilde tanımlanır.

$$T * v_{ij} = \sum_{C(k,1) \in N_r(i,j)} T(k-1,1-j) v_{kl}$$

$T(m,n)$ küme kalıbı içinde m . satır ve n . sütunda bulunan elemanı gösterir. $(r = 1)$ komşuluğunda $m = n = -1, 0, 1$ değerlerini almaktadırlar.

Yukarıdaki tanımda $T(i,j;k,1)$ değerinin i ve j 'den bağımsız olarak ifade edilebileceği ortaya konulmuştur. Küme kalıplarına bu özellikleri nedeniyle uzayda değişmeyen (space invariant) matrisler denir. Bu özellik, hücrel ağların dinamik kurallarını küme kalıplarına dayanarak verme imkanı sağlamaktadır.

3.3 Kanonik Hücre Denklemleri

Hücre içindeki tüm kaynaklar gerilim kontrollü veya sabit kaynaklar olduğundan ve devrede u , x , y olmak üzere üç düğüm bulunduğundan, devre düğüm analizine uygundur. u , x ve y için düğüm denklemleri yazılırsa:

$$(u) \quad v_{uij} - E_{ij} = 0 \quad (3.1a)$$

$$(y) \quad v_{yij}(t) - I_{yx} \cdot R_y = 0 \quad (3.1b)$$

$$(x) \quad -C \frac{dv_{xij}(t)}{dt} + I - \frac{1}{R_x} v_{xij}(t) + A * v_{yij}(t) + B * v_{uij} = 0 \quad (3.1c)$$

Düğüm denklemleri, durum denklemlerini verecek şekilde düzenlenirse; aşağıdaki kanonik devre denklemlerini elde ederiz:

$$C \frac{dv_{xij}(t)}{dt} = - \frac{1}{R_x} v_{xij}(t) + A * v_{yij}(t) + B * v_{uij} + I \quad (3.2a)$$

$$v_{yij}(t) = \frac{1}{2} \left[|v_{xij}(t)+1| - |v_{xij}(t)-1| \right] \quad (3.2b)$$

$$v_{uij} = E_{ij} \quad \begin{matrix} 1 \leq i \leq M \\ 1 \leq j \leq N \end{matrix} \quad (3.2c)$$

Chua tarafından hücresel ağ teorisi gerçekleştirilirken değişkenler ve parametreler üzerine bazı kısıtlar getirilmiştir. Bunlar aşağıda verilmiştir.

$$|v_{xij}(0)| \leq 1 \quad (3.2d)$$

$$|v_{uij}| \leq 1 \quad (3.2e)$$

$$AC(i,j;k,1) = AC(k,1;i,j) \quad \text{Karşılıklık (reciprocal)}$$

$$C > 0, \quad R_x > 0$$

Kısıtlamalardan bazıları, zamanla çalışmalar ilerledikçe yumuşatıldıysa da, gene de teoremin büyük bir kısmı bu kısıtlara dayanılarak ortaya atılmıştır. Örneğin ilk başta küme kalıplarının karşılıklı özelliğini sağlaması istenirken, 1990 yılında Chua ve Roska tarafından yayınlanan bir makalede, karşıt olmayan küme kalıplarının da belirli koşullar altında yakınsak olduğu kanıtlanmıştır.

(3.2) denklemlerine dayanarak bir devre tasarlamadan önce, yapacağımız devrenin durumlarının ve çıkışlarının, yani dinamik cevabının sınırlı olduğundan emin olmamız gerekir. Aşağıdaki teorem, hücresel ağların dinamik cevabının sınırlı olduğunu göstermektedir.

Teorem 3.1: Bir hücresel nöral ağda bütün durumlar $t > 0$ için sınırlıdır ve bu $v_{\max}$ sınırı aşağıdaki formülle hesaplanır.

$$v_{\max} = 1 + R_x |I| + R_x \max \left[\sum_{C(k,1) \in N_r(1,j)} \left[|AC(i,j;k,1)| + |BC(i,j;k,1)| \right] \right] \quad (3.3)$$

Tanıt : Hücreye ait dinamik denklemi gözönüne alırsak

$$\frac{dv_{xij}(t)}{dt} = -\frac{1}{R_x} v_{xij}(t) + \frac{1}{C} A * v_{yij}(t) + \frac{1}{C} B * v_{uij}(t) + \frac{1}{C} I \quad (3.4)$$

Bu denklemlerde şu değişiklikleri yapalım :

$$\frac{1}{C} A * v_{yij}(t) \longrightarrow f_{ij}(t)$$

$$\frac{1}{C} B * v_{uij} \longrightarrow g_{ij}(t)$$

$$\frac{1}{C} I \longrightarrow \hat{I}$$

Buradan elde edilen denklem

$$\frac{dv_{xij}(t)}{dt} = -\frac{1}{R_x} v_{xij}(t) + f_{ij}(t) + g_{ij}(t) + \hat{I} \quad (3.5)$$

şeklinde 1. dereceden bir adi diferansiyel denklemdir ve çözümü kolayca yazılabilir.

$$v_{xij} = v_{xij}(0) \cdot e^{-t/R_x C} + \int_0^t e^{-(t-\tau)/R_x C} \left[f(\tau) + g(u) + \hat{I} \right] d\tau \quad (3.6)$$

Amacımız v_{xij} için bir üst sınır bulmak olduğundan:

$$|v_{xij}| \leq \left| v_{xij}(0) \cdot e^{-t/R_x C} \right| + \left| \int_0^t e^{-(t-\tau)/R_x C} \left[f(\tau) + g(u) + \hat{I} \right] d\tau \right| \quad (3.7)$$

$$|v_{xij}| \leq |v_{xij}(0)| e^{-l/R_x C} + \int_0^l e^{-(l-\tau)/R_x C} \left| \left[f(\tau) + g(u) + \hat{I} \right] \right| d\tau \quad (3.8)$$

$$|v_{xij}| \leq |v_{xij}(0)| e^{-l/R_x C} + \int_0^l e^{-(l-\tau)/R_x C} \left[|f(\tau)| + |g(u)| + |\hat{I}| \right] d\tau \quad (3.9)$$

Eşitsizliğin yönüne uygun olarak $f(\tau)$ ve $g(u)$ fonksiyonlarını içerdikleri en büyük değerle yer değiştirelim.

$$\begin{aligned} F_{ij} = \max_t |f_{ij}(t)| &\leq \frac{1}{C} \sum_{c(k,l) \in N_r(i,j)} |A(i,j,k,l)| v_{ykl}(t) \\ &\leq \frac{1}{C} \sum_{c(k,l) \in N_r(i,j)} |A(i,j,k,l)| \cdot \max_t |v_{ykl}(t)| \end{aligned} \quad (3.10)$$

$$\begin{aligned} G_{ij} = \max_u |g_{ij}(u)| &\leq \frac{1}{C} \sum_{c(k,l) \in N_r(i,j)} |B(i,j,k,l)| v_{ukl} \\ &\leq \frac{1}{C} \sum_{c(k,l) \in N_r(i,j)} |B(i,j,k,l)| \cdot \max_u |v_{ukl}| \end{aligned} \quad (3.11)$$

Bu değerleri denklemde yerlerine koyarsak,

$$|v_{xij}(t)| \leq |v_{xij}(0)| e^{-l/R_x C} + \left[F_{ij} + G_{ij} + \hat{I} \right] \int_0^l e^{-(l-\tau)/R_x C} d\tau \quad (3.12)$$

$$\int_0^l e^{-(l-\tau)/R_x C} d\tau = R_x C \cdot \left[1 - e^{-l/R_x C} \right]$$

$$|v_{xij}(t)| \leq |v_{xij}(0)| e^{-\frac{1}{R_x} C} + R_x C [F_{ij} + G_{ij} + \hat{I}] \left[1 - e^{-\frac{1}{R_x} C} \right] \quad (3.13)$$

Her iki terimin de maksimum değerleri alınırsa ;

$$|v_{xij}(t)| \leq |v_{xij}(0)| + R_x C [F_{ij} + G_{ij} + \hat{I}] \quad (3.14)$$

$$|v_{xij}(t)| \leq |v_{xij}(0)| + R_x \left[\sum_{c(k,l) \in N_r(i,j)} |AC(i,j,k,l)| \cdot \max_t |v_{ykl}(t)| + \sum_{c(k,l) \in N_r(i,j)} |BC(i,j,k,l)| \cdot \max_u |v_{ukl}| + \hat{I} \right] \\ |v_{xij}(0)| \leq 1, \quad |v_{uij}| \leq 1, \quad |v_{yij}(t)| \leq 1 \quad (3.15)$$

$$|v_{xij}(t)| \leq 1 + R_x \hat{I} + R_x \left[\sum_{c(k,l) \in N_r(i,j)} |AC(i,j,k,l)| + |BC(i,j,k,l)| \right] \quad (3.16)$$

Üst sınırı elde edilir.

3.4 Enerji Fonksiyonu ve Denge Durumları

Nöral ağlarda sürekli olarak karşımıza karmaşık diferansiyel denklem takımları çıkmaktadır. Diferansiyel denklem takımlarını matematiksel olarak çözmek her zaman

parça-lineer diferansiyel denklemler çıkmaktadır. Devre teorisinde, bir sistemin kararlı olup olmadığının belirlenebilmesi için en çok kullanılan yöntem, sisteme Liapunov anlamında kararlılık kriterinin uygulanmasıdır. Liapunov Teoreminin en önemli özelliği, bir diferansiyel denklemi çözmeden, uygun bir fonksiyonun varlığından yararlanarak yakınsaklığını gösterebilmesidir. Özellikle lineer-olmayan fonksiyonlar içeren diferansiyel denklemlerin analitik yöntemlerle çözümü olanaksız olduğunda, Liapunov Teoremine başvurulur.

Liapunov Kararlılık Kriterini genel olarak şöyle ifade edebiliriz : Eğer bir sistem asimptotik olarak bir denge durumuna yakınsıyorsa, bu sistemde toplanan enerji zamanla azalır ve denge durumunda da minimum değerini alır.

Bir diferansiyel denklem sisteminin Liapunov anlamında kararlı olduğunu göstermek için önce bu denklemin yörüngeleri boyunca aşağıdaki özelliklere sahip bir fonksiyonun bulunması gerekir:

Liapunov Fonksiyonu: $v : \mathbb{R}^N \times \mathbb{R} \rightarrow \mathbb{R}$ olmak üzere $v(x,t)$ skaler fonksiyonunun orjin etrafındaki kesin-pozitif Ω bölgesi için hem kendisi hem de argümanlarına göre türevleri kesin-pozitif ise ve yörünge boyunca zamana göre türevi kesin-negatif ise bu fonksiyona diferansiyel denklemin Liapunov fonksiyonu denir.

Liapunov fonksiyonuna enerji fonksiyonu da denmesine rağmen fiziksel olarak devrenin enerjisi ile kesin bir bağlantısı kurulamamıştır. Bir sisteme ait pek çok Liapunov fonksiyonu bulunabilir. Özellikle lineer olmayan sistemlerde Liapunov fonksiyonlarının bulunması için hiçbir özel yöntem verilmemiştir. Fonksiyonu bulmak kişinin görüş yeteneğine ve deneyimine kalmıştır [19].

Liapunov Teoremi: $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t)$

$$\mathbf{f}(0, t) = 0$$

olarak verilmiş bir sisteme ait kesin-pozitif ve

a) Sürekli 1.kısmi türevleri olan

b) Durum değişkeninin yörüngeleri boyunca türevi $\dot{V}(\mathbf{x}, t)$ kesin pozitif olan bir $V(\mathbf{x}, t)$ fonksiyonu bulunabilir ise orjinde bulunan denge noktası asimptotik olarak kararlıdır.

Bir diferansiyel denklem sisteminin kararlılığını göstermek için yukarıdaki koşulları sağlayan bir Liapunov fonksiyonu bulmak yeterlidir. Hücre sel nöral ağlarda ise Liapunov fonksiyonu şöyle tanımlanır :

$$\begin{aligned} E(t) = & -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{\substack{c(k,l) \in \\ r}} AC(i,j;k,l) v_{yij}(t) v_{ykl}(t) \\ & + \frac{1}{2R_x} \sum_{i=1}^N \sum_{j=1}^M v_{yij}^2(t) - \sum_{i=1}^N \sum_{j=1}^M \sum_{\substack{c(k,l) \in \\ r}} BC(i,j;k,l) v_{yij}(t) v_{ukl} \\ & - \sum_{i=1}^N \sum_{j=1}^M I v_{yij}(t) \end{aligned} \quad (3.17)$$

Yukarıdaki enerji fonksiyonu, hücre içindeki lineer olmayan kaynağın parça-lineer karakteristiğine göre bulunmuştur. Oysa gerçek hayatta devre elemanları daha yumuşak geçişleri olan sürekli sigmoidal fonksiyon karakteristiklerine sahiptirler. Hücrede sigmoidal karakteristiğe sahip devre elemanı kullanıldığında Liapunov fonksiyonu aşağıdaki şekli alır .

$$\begin{aligned}
 E(t) &= -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{C(k,l) \in N_r(i,j)} A(i,j;k,l) v_{yij}(t) v_{ykl}(t) \\
 &- \frac{1}{2R_x} \sum_{i=1}^N \sum_{j=1}^M \int_0^{v_{yij}(t)} f^{-1}(\omega) d\omega \\
 &- \sum_{i=1}^N \sum_{j=1}^M \sum_{C(k,l) \in N_r(i,j)} B(i,j;k,l) v_{yij}(t) v_{ykl} \\
 &- \sum_{i=1}^N \sum_{j=1}^M I v_{yij}(t) \tag{3.18}
 \end{aligned}$$

Hücresel ağlarda yakınsaklığı belirledikten sonra kararlı hal denge durumlarının da bulunması gerekir.

Tanım 3.5: Bir hücresel ağda çıkış gerilimleri

$$v_{yij} = \mp 1 \quad \forall C(k,l) \in N_r(i,j)$$

konumunda iken

$$\left. \frac{dv_{xij}(t)}{dt} \right|_{v_{xij}=v_{xij}^*} = 0 \quad \text{ve} \quad |v_{xij}^*| < 1$$

koşulunu sağlayan v_{xij}^* durumuna kararlı hücre denge durumu denir.

Tanım 3.6: Hücresel nöral ağda, tüm elemanları kararlı hücre denge durumlarından oluşan vektöre kararlı sistem denge durumu denir.

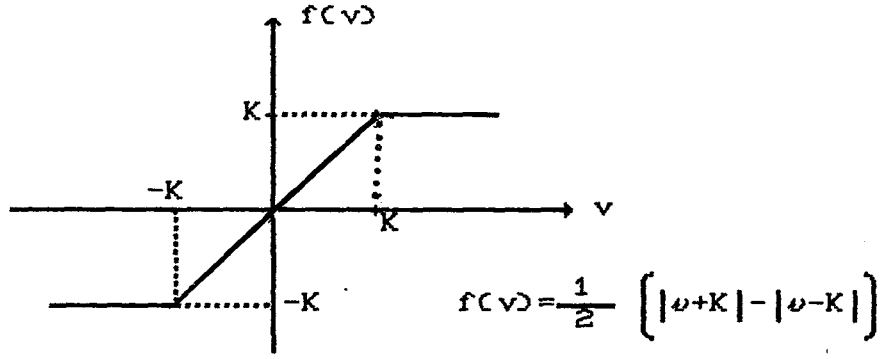
Hücreesel ağların dinamik yapısı, başlangıç noktasından kararlı bir denge durumuna giden yörüngeler üzerine kuruludur. Her kararlı denge noktası, bir grup yörüngenin sonlandığı noktadır ve bu noktaları sistem yörüngelerini içlerine doğru çekmelerinden dolayı etki havuzları da denir. Genel olarak hücreesel ağlar, çıkış uzayını her biri bir grup yörüngenin sonunda yer alan etki havuzlarına bölerler. Giriş uzayı ve çıkış uzayı arasındaki lineer-olmayan bu dönüşüm özellikleri nedeniyle hücreesel ağlar asosiyatif bellek, hata düzeltici kodlama ve hata toleranslı hesaplama gibi geniş uygulama alanlarına sahiptir.

3.5 Lineer-Olmayan ve Gecikmeli Küme Kalıbı Türleri

Hücreesel ağların ilk uygulamalarında sürekli olarak reel sayılardan oluşmuş küme kalıpları kullanılmıştır. Fakat son yıllarda yapılan çalışmalarla, lineer-olmayan ve gecikmeli küme kalıplarının da kullanılabileceği gösterilmiş ve hücreesel nöral ağ kavramı, analog bilgisayar kavramına yaklaştırılmıştır[20]. Lineer-olmayan ve gecikmeli küme kalıplarını kullanmak için [17]'de verilen hücre yapısı, komşu hücrelerle bağlantıyı sağlayan $A(i,j;k,l)$ ve $B(i,j;k,l)$ gerilim kontrollü akım kaynakları,

$$\begin{aligned} \hat{A}_{ijkl}(v_{ykl}, v_{yij}) + A_{ijkl}^T v_{ykl}(t-\tau) \\ \hat{B}_{ijkl}(v_{ukl}, v_{uij}) + B_{ijkl}^T v_{ykl}(t-\tau) \end{aligned}$$

olacak şekilde değiştirilir. $\hat{A}_{ijkl}$ ve $\hat{B}_{ijkl}$ 'deki lineer olmayan fonksiyonun yapısı da önemlidir. Fonksiyonlar en fazla iki değişkene sahip olabilirler. Ayrıca çıkış fonksiyonu da τ_1 değerleri yerine τ_K değerleri arasında tanımlanır ve çıkışların analog seviyeler olması sağlanır (Şekil 3.4).



Şekil 3.4 Genişletilmiş çıkış fonksiyonu

Yeni küme kalıpları kullanılarak kanonik hücre yapısı aşağıdaki gibi elde edilir.

$$\begin{aligned}
 \dot{v}_{xij}(t) = & -\frac{1}{R_x} v_{xij}(t) + \sum_{i,j} \sum_{\alpha(k,l) \in N_r(i,j)} \hat{A}_{ij;kl}(v_{ykl}(t), v_{yij}(t)) \\
 & + \sum_{i,j} \sum_{\alpha(k,l) \in N_r(i,j)} \hat{B}_{ij;kl}(v_{ukl}(t), v_{yij}(t)) \\
 & + \sum_{i,j} \sum_{\alpha(k,l) \in N_r(i,j)} A^T_{ij;kl} v_{ykl}(t-\tau) \\
 & + \sum_{i,j} \sum_{\alpha(k,l) \in N_r(i,j)} B^T_{ij;kl} v_{ukl}(t-\tau) + I \quad (3.20)
 \end{aligned}$$

$$v_{yij}(t) = f(v_{xij}(t)) \quad (3.21)$$

Sistemi daha da geliştirmek istersek çıkış fonksiyonunu da dinamik hale getirebiliriz. Örneğin 3.21 yerine

$$\dot{v}_{yij}(t) = -v_{yij}(t) + f(v_{xij}(t)) \quad (3.22)$$

dinamik denklemi kullanılabilir. Çıkış fonksiyonunun en önemli özelliği monoton artan olmasıdır. Monoton artma koşulu sağlandıktan sonra fonksiyon daha da karmaşık hale getirilebilir. Lineer-olmayan küme kalıbı kullanırken

giriş gerilimi ve başlangıç durumlarına ait (3.2d) ve (3.2e) kısıtları aynen bırakılmalıdır.

Örnek:

$$d_1 = c_1 (e^{v_{ykl}} - 1)$$

$$d_2 = c_2 (v_{ykl} - v_{yij})$$

olsun. d_1 ve d_2 fonksiyonları ile aşağıdaki küme kalıplarını oluşturabiliriz :

$$\hat{A} = \begin{bmatrix} 0 & d_1 & 0 \\ d_1 & 2 & d_1 \\ 0 & d_1 & 0 \end{bmatrix} \quad \hat{B} = \begin{bmatrix} 0 & 0 & 0 \\ d_2 & 1 & d_2 \\ 0 & 0 & 0 \end{bmatrix}$$

Lineer-olmayan küme kalıplarının uygulamaları arasında gene önemli bir alanı görüntü işleme kaplamaktadır. Bazı özel tiplerdeki diferansiyel denklemler de bu yapı ile çözülebilmektedir. Gecikmeli küme kalıplarıyla da ardışıl görüntüler veya hareketli cisimler üzerinde işlemler yapılabilir.

Lineer-olmayan küme kalıbı kullanıldığında hücre potansiyellerinin alacağı maximum değerin,

$$v_{\max} = 1 + R_x |I| + R_x K \cdot \max \left\{ \sum_{c(k,l) \in N_r} \left(\max_{(i,j)} |\hat{A}_{ijkl}| + |\hat{B}_{ijkl}| \right) \right\}$$

(3.23)

olacağı görülebilir. Kalıpların içinde yer alacak lineer-olmayan fonksiyonlar üzerine belirgin bir kısıtlama getirilmediğinden, önermeler henüz teorem haline getirilememiştir.

Küme kalıplarının daha da geliştirilmesiyle hücresele nöral ağlar, analog bilgisayarlara dönüştürülür. Bu nedenle hücresele ağlarla hücresele otomata ve sistolik diziler arasında büyük benzerlikler vardır. Kısaca ifade etmek gerekirse;

	<u>Hücresele Otomata</u>	<u>Sistolik Dizi</u>	<u>HNA</u>
Giriş:	(1-4 Bit) Lojik Değer	(8-32 Bit) Sayısal Değer	Analog Değer
Program:	Doğruluk Tablosu	Sayısal Algoritma	Küme Kalıbı

Hücresele otomata ve sistolik dizilerde devrenin programlanması kolayca yapılabiliyorsa da hücresele ağlarda bu konuda kesin yöntemler elde edilememiştir. Küme kalıpları tamamen devrenin dinamiğini belirlediğinden bu kalıplara analog makro da denilebilir. Gelecekte analog makroların ardışıl veya paralel olarak hücresele ağlara uygulanmasıyla analog algoritmaların elde edileceği ve hücresele ağların programlanmasına açıklık getirileceği umulmaktadır.

BÖLÜM IV

HÜCRESEL NÖRAL AĞLARDA KARARLILIK

Hücresel nöral ağ teorisinin büyük bir kısmını kararlılık analizleri oluşturmaktadır. Devrenin kararlılığının incelenmesinde ilk olarak Liapunov teoreminden yararlanılmış ve karşıt küme kalıplı devrelerde enerji fonksiyonunun monoton azalan bir Liapunov fonksiyonu olduğu ispatlanmıştır. Daha sonraki yıllarda Morris Hirsch tarafından ortaya atılan birleşik ve indirgenemeyen sistemler anlamında da hücresel ağların kararlılığı incelenmiş ve belirli koşullar altında karşıt yapıda olmayan küme kalıplarının da yakınsak olduğu ispatlanmıştır [21]. Yakınsak bir hücresel ağın denge durumlarının bulunması konusunda da çalışmalar yapılmış ve bir algoritma geliştirilmiştir [22].

4.1 Karşıt Küme Kalıplı Ağlarda Kararlılık ve Denge Durumlarının Analizi

1988 yılında Chua ve Lin tarafından hücresel ağ teorisi ortaya atıldığında, küme kalıplarının yapısına önemli bir kısıt getirilmişti. Karşıtlık (reciprocal) olarak adlandırdığımız bu özelliği

$$AC_{i,j;k,l} = AC_{k,l;i,j}$$

şeklinde gösterebiliriz. Küme kalıplarının karşıt olabilmeleri için matrisin ortasındaki eleman $AC_{(0,0)}$ 'a göre simetrik olmaları gerekir. Örneğin

$$A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

A_1 kalıbı karşıt olduğu halde A_2 kalıbı karşıt değildir.

Liapunov kararlılık kriterini kullanarak ağın kararlı olduğunu gösterebiliriz. Bunun için (3.17) de verilen fonksiyonun hücresel nöral ağa ait bir Liapunov fonksiyonu olduğunu göstermemiz yeterlidir. Öncelikle $E(t)$ 'nin sınırlı olduğunu gösterelim.

Teorem 4.1: $M \times N$ boyutlu bir hücresel nöral ağda $E(t)$ enerji fonksiyonu

$$\max |E(t)| \leq E_{\max} \quad (4.1a)$$

değeriyle sınırlıdır ve bu sınır

$$E_{\max} = \frac{1}{2} \sum_{i,j} \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} |AC(i,j;k,l)| + \sum_{i,j} \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} |BC(i,j;k,l)| \\ + MN \left[\frac{1}{2R_x} + |I| \right] \quad (4.1b)$$

ifadesinden hesaplanabilir.

Tanıt: Enerji fonksiyonunun tanımından

$$E(t) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{c(k,l) \in N_r(i,j)} AC(i,j;k,l) v_{yij}(t) v_{ykl}(t) \\ + \frac{1}{2R_x} \sum_{i=1}^N \sum_{j=1}^M v_{yij}^2(t) - \sum_{i=1}^N \sum_{j=1}^M \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} BC(i,j;k,l) v_{yij}(t) v_{ukl}(t) \\ - \sum_{i=1}^N \sum_{j=1}^M I_i v_{yij}(t)$$

$$\begin{aligned}
 |E(t)| \leq & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} |AC(i,j;k,l)| \cdot |v_{yij}(t)| \cdot |v_{ykl}(t)| \\
 & + \frac{1}{2R_x} \sum_{i=1}^N \sum_{j=1}^M v_{yij}^2(t) + \sum_{i=1}^N \sum_{j=1}^M |I| \cdot |v_{yij}(t)| \\
 & + \sum_{i=1}^N \sum_{j=1}^M \sum_{c(k,l) \in N_r(i,j)} |BC(i,j;k,l)| \cdot |v_{yij}(t)| \cdot |v_{ukl}|
 \end{aligned} \tag{4.3}$$

Bütün giriş ve çıkış değerleri yerine maximum değer +1 koyulursa,

$$\begin{aligned}
 |E(t)| \leq & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{c(k,l) \in N_r(i,j)} |AC(i,j;k,l)| \\
 & + \sum_{i=1}^N \sum_{j=1}^M \sum_{c(k,l) \in N_r(i,j)} |BC(i,j;k,l)| + MN \left[\frac{1}{2R_x} + |I| \right]
 \end{aligned} \tag{4.4}$$

olarak $E(t)$ 'nin üst sınırı elde edilir.

Teorem 4.2: Enerji fonksiyonu $E(t)$

$$\frac{dE(t)}{dt} \leq 0$$

olacak şekilde monoton azalan bir fonksiyondur.

Tanıt: Tanıt sırasında $v_{yij}(t)$ 'nin türevi yerine

$$\frac{dv_{yij}(t)}{dt} = \frac{dv_{yij}(t)}{dv_{xij}(t)} \cdot \frac{dv_{xij}(t)}{dt}$$

ifadesini kullanalım. Zamana göre $E(t)$ 'nin türevini alırsak,

$$\begin{aligned}
 \frac{dE(t)}{dt} = & - \sum_{i=1}^N \sum_{j=1}^M \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} A(i,j;k,l) \frac{dv_{yij}(t)}{dv_{xij}(t)} \frac{dv_{xij}(t)}{dt} v_{ykl}(t) \\
 & + \frac{1}{R_x} \sum_{i,j} \frac{dv_{yij}(t)}{dv_{xij}(t)} \cdot \frac{dv_{xij}(t)}{dt} v_{yij}(t) \\
 & - \sum_{i=1}^N \sum_{j=1}^M \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} B(i,j;k,l) \frac{dv_{yij}(t)}{dv_{xij}(t)} \frac{dv_{xij}(t)}{dt} v_{ukl} \\
 & - \sum_{i,j} I \cdot \frac{dv_{yij}(t)}{dv_{xij}(t)} \frac{dv_{xij}(t)}{dt} \quad (4.6)
 \end{aligned}$$

Yukarıdaki ifadede küme kalıplarınının karşıtlık özelliği kullanılmıştır. Çıkış fonksiyonu (3.2b) gözönüne alınırsa, türevler

$$\frac{dv_{yij}(t)}{dt} = \begin{cases} 1 & , \quad |v_{xij}| < 1 \\ 0 & , \quad |v_{xij}| \geq 1 \end{cases} \quad (4.7)$$

$$\begin{aligned}
 \frac{dE(t)}{dt} = & - \sum_{i,j} \frac{dv_{yij}(t)}{dv_{xij}(t)} \frac{dv_{xij}(t)}{dt} \left[\sum_{\substack{c(k,l) \in \\ N_r(i,j)}} A(i,j;k,l) v_{ykl}(t) \right. \\
 & \left. - \frac{1}{R_x} v_{yij}(t) + \sum_{\substack{c(k,l) \in \\ N_r(i,j)}} B(i,j;k,l) v_{ukl} + I \right] \quad (4.8)
 \end{aligned}$$

$$= - \sum_{\substack{(i,j) \\ |v_{xij}| < 1}} \frac{dv_{xij}(t)}{dt} \left[\sum_{\substack{c(k,l) \in \\ N_r(i,j)}} AC(i,j;k,l) v_{ykl}(t) \right. \\ \left. - \frac{1}{R_x} v_{xij}(t) + \sum_{c(k,l) \in N_r(i,j)} BC(i,j;k,l) v_{ukl} + I \right] \quad (4.9)$$

olarak elde edilir. Hücre durum denklemi

$$C \frac{dv_{xij}(t)}{dt} = - \frac{1}{R_x} v_{xij}(t) + A * v_{yij}(t) + B * v_{uij} + I \quad (4.10)$$

şeklinde olduğundan, (4.9) ifadesinde parantezin içi (4.10) ifadesinde eşitliğin sağ tarafına eşittir.

Buradan

$$\begin{aligned} \frac{dE(t)}{dt} &= - \sum_{\substack{(i,j) \\ |v_{xij}| < 1}} C \left[\frac{dv_{xij}(t)}{dt} \right]^2 = - \sum_{\substack{(i,j) \\ |v_{xij}| < 1}} C \left[\frac{dv_{yij}(t)}{dt} \right]^2 = \\ &= - \sum_{(i,j)} C \left[\frac{dv_{yij}(t)}{dt} \right]^2 \leq 0 \end{aligned}$$

dır. Teorem 4.1 ve 4.2'den aşağıdaki sonuca varabiliriz.

Teorem 4.3: Bir hücresel nöral ağda herhangi bir v_u girişi vektörü ve v_x başlangıç durum vektörü verildiğinde

$$\lim_{t \rightarrow \infty} E(t) = \text{sabit} \quad \text{ve} \quad (4.11a)$$

$$\lim_{t \rightarrow \infty} \frac{dE(t)}{dt} = 0 \quad (4.11b)$$

olur.

Tanıt: Teorem 4.1 ve 4.2'nin sonucuna göre $E(t)$ sınırlı

ve monoton azalan bir fonksiyondur. Bu durumda bir limiti vardır ve limit noktasında türevi sifıra eşittir.

$E(t)$ 'nin monoton azalan, sürekli bir fonksiyon olması bir Liapunov fonksiyonu olduğunu gösterir. $E(t)$ 'nin bir limiti vardır ve bu limite azalarak yaklaştığı için limit noktası aynı zamanda fonksiyonun minimum noktasıdır. Bu özelliklerinden dolayı $E(t)$ 'nin tanımladığı sistem kararlıdır diyebiliriz.

Yukarıdaki teoremlerin ışığında hücre çıkışlarına ait önemli bir özelliği de söyleyebiliriz. Hücresel ağın geçiş durumu söndükten sonra daima sabit bir çıkış gerilimi elde edilir. Açıkça ifade edilirse ;

$$\lim_{t \rightarrow \infty} v_{yij}(t) = \text{sabit} \quad (4.12a)$$

$$1 \leq i \leq M$$

$$1 \leq j \leq N$$

$$\lim_{t \rightarrow \infty} \frac{dv_{yij}(t)}{dt} = 0 \quad (4.12b)$$

dır.

Ağın kararlı hale ulaştıktan sonraki durumu hakkında da bazı tahminlerde bulunabiliriz. Çıkış fonksiyonu $v_{yij}(t)$ 'nin parça sürekli yapısı nedeniyle, kararlı halde durum değişkenleri $v_{xij}(t)$ 'ler şu değerleri alabilirler:

Kararlı halde

$$\frac{dE(t)}{dt} = 0 \Rightarrow \frac{dE(t)}{dt} = - \sum_{(i,j)} c \left[\frac{dv_{yij}(t)}{dt} \right]^2 = 0$$

$$1) \quad |v_{xij}| < 1 \Rightarrow v_{yij}(t) = v_{xij}(t) \Rightarrow \frac{dv_{xij}(t)}{dt} = 0$$

$|v_{xij}| \geq 1 \Rightarrow v_{xij}(t)$ nin durumu hakkında kesin bir yargıda bulunulamaz.

$$2) \frac{dv_{xij}(t)}{dt} = 0 \Rightarrow v_{xij}(t) \text{ sabit bir fonksiyondur.}$$

$$3) \frac{dv_{xij}(t)}{dt} \neq 0 \Rightarrow v_{xij}(t) \text{ zamanla deęişen fakat sınırlı bir fonksiyon olabilir.}$$

Hücreesel nöral ağlarda devre parametreleri $AC_{i,j;i,j}$ ve R_x üzerinde bazı koşullar koyarak, kararlı halde durum deęişkenlerinin sadece 1. ve 2. durumlarda olabileceğini söyleyebiliriz.

Teorem 4.4: Bir hücreesel nöral ağda devre parametreleri

$$AC_{i,j;i,j} > 1/R_x$$

koşulunu sağlarsa, devrenin geçiř durumu söndükten sonra bütün durum deęişkenleri bir kararlı hücre denge durumuna yerleşir ve genlikleri de 1'den büyük olur. Matematiksel olarak ifade edersek;

$$\lim_{t \rightarrow \infty} |v_{xij}(t)| > 1$$

$$\lim_{t \rightarrow \infty} v_{yij}(t) = \mp 1$$

Tanıt için [17] numaralı referansa başvurulabilir.

4.2 Birlikte Çalışan ve İndirgenemeyen Sistemler

Hangi amaçla olursa olsun, bir nöral ağ tasarlama aşamasında devrenin yakınsak olup olmadığını göstermek çok önemlidir. Yakınsaklığın da bazı çeşitlerinden söz edebiliriz. Başlangıç değerine de baęlı olarak, her ileri yörüngenin kararlı bir durumla sonuçlandığı sistemlere yakınsak sistemler denir. Daha zayıf bir ifadeyle hemen hemen her yörüngenin kararlı bir durumla sonuçlandığı sistemlere de hemen hemen yakınsak sistemler

denir. Daha da zayıf bir tanım, her yörüngenin asimptotik olarak kararlı denge durumlarının sınırlı bir kümesine yakınsadığı durumdur ve yarıyakınsak adını alır. Benzer olarak hemen hemen yarıyakınsak tanımı da yapılabilir. Lineer-olmayan diferansiyel denklem sistemlerinin yakınsak olup olmadığının belirlenmesinde önemli bir yöntem olan Liapunov teoreminden daha önce de bahsedilmişti. Yöntemin en önemli dezavantajı, Liapunov fonksiyonlarının bulunması için belirli bir algoritmanın verilmemesidir. Grossberg

$$\dot{x}_i = a_i(x) \left[b_i(x_i) - C(x_1, x_2, \dots, x_n) \right] \quad (4.14)$$

şeklindeki yarışan sistemler sınıfı için bir yakınsaklık teoremi tanımlamış ve b_i 'ler parça-parça-sürekli fonksiyonlar ise sistemin yakınsak olacağını tanıtlamıştır [23]. Hirsch ise 1982-1987 yılları arasında yayınladığı bir dizi makale ile nöral ağların yakınsaklığının belirlenmesi için bir yöntem geliştirmiştir. Hirsch tarafından geliştirilen bu yöntemde bazı tanımlar yapılmıştır.

Tanım 4.1: n boyutlu x vektörü üzerinde tanımlanmış $F(x)$ vektör alanının $DF(x)$ Jacobien matrisinde, ayırık her i ve j indisi arasında $i=k_1, k_2, \dots, k_m=j$ olacak şekilde

$$\frac{\partial F_{k_r}}{\partial x_{k_{r-1}}} \neq 0 \quad r=1, 2, \dots, m \quad (4.15)$$

koşulunu sağlayacak şekilde bir k_r indisi zinciri bulunabiliyorsa F vektör alanına indirgenemez denir.

Nöral ağlarda indirgenemezlik özelliği ağ içindeki her bir nöronun birdiğerini dolaylı da olsa etkilediğini gösterir.

Tanım 4.2: F vektör alanına ait $DF(x)$ Jacobien matrisinde x 'den bağımsız olarak köşegen dışındaki elemanların işaretleri karşılıklı olarak aynı ise sisteme işaret simetrik denir. İşaret simetrik matrislerde

$$\operatorname{sgn} \left[\frac{\partial F_i}{\partial x_j} \right] = \operatorname{sgn} \left[\frac{\partial F_j}{\partial x_i} \right]$$

karşılıklı terimlerin işaretleri aynıdır.

Tanım 4.3: İşaret simetrik bir F vektör alanına ait Jacobien matrisi içindeki bütün terimlerin işaretleri negatif olan sistemlere yarışan, pozitif olan sistemlere birlikte çalışan sistemler denir [24]. Matematiksel olarak ifade edersek

$$\frac{dx_i}{dt} = F_i(x_1, x_2, \dots, x_n) \quad i=1, 2, \dots, n \quad (4.16)$$

şeklinde tanımlanmış sistemlerin Jacobien matrislerinde

$$\frac{\partial F_i}{\partial x_j} \geq 0 \quad i \neq j \quad (4.17a)$$

ise birlikte çalışan,

$$\frac{\partial F_i}{\partial x_j} \leq 0 \quad i \neq j \quad (4.17b)$$

ise yarışan sistemler adını alırlar.

Teorem 4.5: Birlikte çalışan sistemlerin sürekli zaman çözümleri sabit olmayan, periyodik çözümler olamaz.

Teorem 4.6: Eğer sistem birlikte çalışan ve indirgenemez ise yaklaşık her başlangıç koşulu kararlı bir denge durumuna yakınsar.

Her iki teoremin tanıtları da oldukça karmaşık olduğundan buraya alınmamıştır, ancak [25] numaralı kaynaktan bulunabilirler.

Yukarıdaki iki teorem ile nöral ağların yakınsaklık analizlerine yeni bir bakış getirilmiş ve sonuçları ilerdeki yıllarda daha da genişletilerek hücresel nöral ağlara da uygulanmıştır.

4.3 Karşıt Olmayan Küme Kalıplarının Kararlılığı

1990 yılında yayınlanan bir makalede Chua ve Roska, karşıt olmayan küme kalıplarının da bazı koşullar altında karalı olduklarını tanıtladılar. Uygulamada, karşıt küme kalıplarıyla tanıtlanan yakınsaklık fazla önem taşımıyordu, çünkü gerek kullanılan elemanların toleransları, gerekse devrede yaşlanmayla meydana gelen bozulmalar, bağlantı ağırlıklarının tam simetrik olmasını engelliyordu. Burada karşıt olmayan kalıpların yakınsaklığını incelerken bazı tanımlara ihtiyacımız olacaktır.

Tanım 4.6: Aşağıdaki iki koşulu sağlayan küme kalıplarına pozitif geri besleme kalıbı denir.

$$a) AC(i,j;k,l) \geq 0 \quad \forall C(k,l) \in \mathcal{C}(i,j) \text{ ve } i,j \neq k,l \quad (4.18)$$

b) A küme kalıbının elemanları, ağı içinde herhangi bir nörondan diğerine tamamen pozitif katsayılardan oluşmuş bir yol oluşturabilecek şekilde yerleştirilmiş olmalıdırlar. Bu özelliği sağlayan küme kalıplarına hücre-bağılantılı kalıplar denir.

Örneğin şekil 4.1 (a)'daki küme kalıbı hücre-bağılantılı iken, (b)'deki değildir.

Tanım 4.6'nın (a) maddesindeki eşitsizlik ters yöne çevrilirse, bu sefer negatif geribesleme kalıbının tanımını elde edilir, fakat $AC(i,j;i,j) > 0$ koşulu gene devam eder.

Tanım 4.7: Aşağıdaki işaret yapısını yatay, düşey veya köşegenlerinde bulunduran küme kalıplarına ters işaretli kalıplar denir.

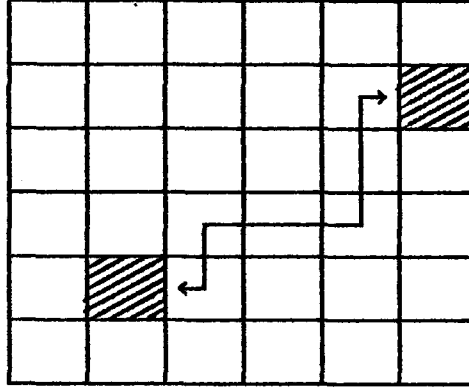
$$A = \begin{bmatrix} 0 & 0 & 0 \\ s & -p & s \\ 0 & 0 & 0 \end{bmatrix} \quad p > \frac{1}{R_x} \quad s \neq 0$$

0	1	0
1	2	1
0	1	0

(a)

0	0	0
0	2	1
0	0	0

(b)



(c)

Şekil 4.1. Hücre-bağlantılı kalıp özelliği.

- a) Hücre-bağlantılı,
- b) Hücre-bağlantılı değil,
- c) Bağlantı biçimi.

Chua ve Roska, pozitif hücre-bağlantılı küme kalıplarının yakınsaklığını bu sistemin birlikte çalışan ve indirgenemeyen bir sistem olduğunu göstererek, aşağıdaki teoremlerle tanıtlamıştır.

Teorem 4.7: (3.2)'de verilen dinamik durum denklemleri ile tanımlanmış, pozitif kalıplı ve $f(\cdot)$ yumuşak geçişli çıkış fonksiyonuna sahip bir hücresel nöral ağ, verilen herhangi $\mathbf{v}_u$ giriş vektörü ve $\mathbf{v}_x$ başlangıç konumu ile, sıfır olasılıklı bir küme dışında tamamen yakınsaktır.⁴ Sıfır olasılıklı kümeden kasıt, sistemi yakınsak olmayan duruma sürükleyen, fakat meydana gelmesi fiziksel sistemlerdeki gürültü nedeniyle olanaksız olan başlangıç koşullarının varlığıdır.

⁴Tamamen yakınsaklık; sistemde her yörüngenin kararlı bir denge noktasında son bulduğunu ifade eder.

Tanıt: (3.2a) ve (3.2b)'de verilen durum ve çıkış denklemlerini yeni bir forma sokmak için durum değişkeni $v_{xij}(t)$ 'yi $M \times N$ boyutlu bir vektör haline getirelim.

$v_{xij}(t)$ 'nin $\tilde{x}$ içinde sütun bazında sıralandığını varsayalım. Durum denklemini yeni $\tilde{x}$ vektörü ile yazmak istediğimizde v_{uij} ve v_{yij} de vektörel yapıya gireceklerdir.

Sonuçta

$$\dot{\tilde{x}} = F(\tilde{x}) = -\tilde{x} + A'f(\tilde{x}) + B'u + I \quad (4.18)$$

denklemini elde ederiz. A' ve B' matrisleri A ve B kalıplarının özelliklerini taşıyan $MN \times MN$ boyutlu matrisler ve I 'da elemanları I olan $MN \times 1$ boyutlu vektördür. A' ve B' matrislerinin köşegenlerinde, küme kalıplarının merkez elemanları bulunacaktır. (4.18)'nin Jacobien matrisi,

$$J = -U + A'.P \quad (4.19)$$

şeklindedir. U birim matris, P ise elemanları $f(.)$ fonksiyonunun eğimlerinden oluşan pozitif diyagonal matristir. Bu durumda Jacobien matrisinin köşegen dışındaki elemanları A' matrisi ile beslenir. A' , pozitif küme kalıbı A 'nın devredeki yerleşiminden elde edildiğinden köşegen dışındaki elemanları pozitifdir ve bu durumda $F(\tilde{x})$ vektör alanı birlikte çalışandır denir.

Kullanılan pozitif küme kalıbının diğer bir özelliği de hücre-bağılantılı olması idi. Bu özelliğe dayanarak, $F(\tilde{x})$ vektör alanının indirgenemeyen yapıya sahip olduğunu söyleyebiliriz.

Teorem 3.1'de bir hücreyel ağda bütün durum gerilimlerinin sınırlı olduğu ve bu sınırın da karşılıklı koşulundan bağımsız olarak (3.3) ifadesiyle hesaplanabileceği tanıtlanmıştı. Buradan $\mathcal{W}^C \subset \mathbb{R}^n$ ve (3.2d) koşulunu sağlamak üzere $\tilde{x}(0) = \tilde{x}_0 \in \mathcal{W}^C$ başlangıç koşullarının bütün ileri yörüngelerinin sınırlı olduğunu

söyleyebiliriz.

Sonuç olarak elimizde indirgenemeyen, birlikte çalışan ve $x_0 \in \mathcal{W}^C \subset \mathbb{R}^n$ başlangıç koşulundan başlayan bütün ileri yörüngeleri sınırlı bir sistem vardır. Teorem 4.6 bağlamında bu sistem yakınsaktır denir.

Teorem 4.7 çeşitli sınıflardan kalıpların yakınsaklığının tanıtlanmasında bir örnek olarak kullanılabilir. Örneğin sadece 1 satır, sütun veya köşegende sıfırdan farklı elemanı olan negatif kalıpların yakınsaklığı veya lineer olmayan küme kalıplarının da devrenin yakınsaklığını etkilemeyeceği tanıtlanabilir.

4.4 Ters İşaretli Küme Kalıplarının Yakınsaklığı

Ters işaretli küme kalıbı tanımı, tanım 4.7'de verilmişti. Ters işaretli küme kalıpları pozitif veya negatif küme kalıbı tanımına uymadıkları halde, sıfırdan farklı elemanların sadece tek doğrultuda olmasından dolayı kararlı bir yapı gösterirler. Durum denklemi (4.8), bir ters işaretli kalıbın kullanılması halinde ($R_x=1, C=1$) aşağıdaki basit formu alır.

$$\dot{\tilde{x}} = -U \tilde{x} + A' f(\tilde{x}) - b \quad (4.20)$$

Ters işaretli satır kalıbı olarak

$$A = \begin{bmatrix} 0 & 0 & 0 \\ s & p & -s \\ 0 & 0 & 0 \end{bmatrix} \quad p > 1, \quad s > 0$$

kullanılırsa A^1 matrisi,

$$A^1 = \begin{bmatrix} p & -s & & & & \\ & s & p & -s & & \\ & & s & p & -s & \\ & & & & & \ddots \\ & & & & & & p & s \\ & & & & & & -s & p \end{bmatrix}$$

ve

(4.21)

$$b_{\sim} = I_{\sim} + \sum_{c(k,b) \in N_r(i,j)} B(i,j;k,l) \cdot v_{ukl}$$

formunu alır. $s < 0$ olması durumunda da hücrelerin sırasını ters çevirmekle gene A^1 formunda bir matris elde edebiliriz. Köşegen tipli kalıplar kullanıldığında ise A^1 'in boyutu, devrede gerçek köşegen üzerindeki elemanların sayısına bağlı olarak 1 ile N arasında değişir. Ters işaretli küme kalıplarının yakınsaklığını incelemek için aşağıdaki yardımcı teoremi kullanacağız.

Lemma 4.1: $P, P \subset \mathbb{R}^{N \times N}$ olmak üzere, bütün temel minörleri pozitif olan matrislerin kümesi ise ; A^1 (4.21)'deki forma sahip olmak üzere

$$(A^1 - U) \in P \quad (4.22)$$

dir. P kümesindeki matrislerin en önemli özelliği, özdeğerlerinin gerçel kısımlarının daima pozitif olmasıdır.

Tanıt: $(A^1 - U)$ matrisinin köşegen üzerindeki elemanları $q = p-1$ olmak üzere, A^1 ile aynı formda olacağı görülmektedir. Ayrıca bütün temel minörler de birbirleriyle aynı formda olacaklardır. 1., 2. ve 3. temel minörlere bakarsak,

$$\det^1 = q > 0, \quad \det^2 = q^2 + s^2 > 0, \quad \det^3 = q \cdot \det^2 + s^2 \cdot \det^1 > 0$$

bulunur. A^1 matrisinde bütün satırlar aynı formda olduğu için, genelleştirme yoluyla $(k+1)$. minörün

determinantı için

$$\det^{k+1} = q \cdot \det^k + s^2 \cdot \det^{k-1} > 0 \quad (4.23)$$

sonucu bulunur.

Lemma 4.1, küme kalıbı simetrik değerli olmadığı zaman da doğrudur. Yani

$$A = \begin{bmatrix} 0 & 0 & 0 \\ r & p & -s \\ 0 & 0 & 0 \end{bmatrix} \quad p > 1, \quad r, s > 0$$

şeklinde bir küme kullanıldığında da $(A^1 - U) \in P$ özelliği geçerlidir.

Tanım 4.8: (4.20) denkleminde k. durum değişkeninin değeri $|x_k| < 1$ yani $f(x_k) = x_k$ ise hücre α -bölgesinde çalışıyor denir. Benzer şekilde $|x_k| \geq 1$ ise hücre β -bölgesinde çalışıyor denir.

Ters işaretli küme kalıplarının çalışma karakteristiği aşağıdaki teoremle açıklanmıştır.

Teorem 4.8: N, ters işaretli küme kalıplı ve (4.20) dinamik denklemiyle tanımlanmış bir nöral devre olsun. (3.2d) ve (3.2e) kısıtlarına uygun herhangi bir başlangıç koşulu ile verilen devre dinamiğinin aşağıdaki özellikleri vardır.

1) $t=t_j$ anında bütün hücreler α bölgesinde çalışıyorsa, t_j anındaki Jacobien matrisinin bütün özdeğerlerinin pozitif gerçel kısımları vardır.

2) $t=t_j$ anında bütün hücreler β bölgesinde çalışıyorsa, Jacobien matrisinin bütün özdeğerleri -1'dedir ve sistem bu bölgedeki denge noktasına yakınsar.

3) $t=t_j$ anında b adet hücre β bölgesinde, kalanları da α bölgesinde çalışıyorsa, Jacobien matrisinin b adet -1 değerli özdeğeri vardır ve diğer özdeğerlerin gerçel kısımları pozitifdir.

$$A^{\alpha\beta} = \begin{array}{c|ccc|ccc} & & & & k-1 & k & k+1 \\ \hline q & -s & & & & & \\ s & q & -s & & & & \\ & & & & & & \\ & & & & -s & & \\ & & & s & q & 0 & \dots \dots \dots k-1 \\ \hline & & & s & -1 & -s & \dots \dots \dots k \\ & & & & 0 & q & -s \dots \dots \dots k+1 \\ & & & & & s & q & -s \\ & & & & & & \ddots \\ & & & & & & & \ddots \\ & & & & & & & & q \end{array} \quad (4.26a)$$

$$\begin{aligned}
 \tilde{b}_k &= \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{k-2} \\ b_{k-1} - sh \\ b_k + ph \\ b_{k+1} + sh \\ b_{k+2} \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} & q &= p-1 \\
 & & h &= f(x_k) = 1 \text{ veya } -1
 \end{aligned}
 \tag{4.26b}$$

$J=A^{\alpha\beta}$ Jacobien matrisinin özdeğerleri $\det(\lambda U-A^{\alpha\beta})$ nın kökleridir. $A^{\alpha\beta}$ nın köşegen yapısı nedeniyle determinantı şöyle ifade edebiliriz.

$$\det(\lambda U-A^{\alpha\beta}) = \det(\lambda U_{k-1}-A^1) \cdot (\lambda+1) \cdot \det(\lambda U_{n-k}-A^2) \tag{4.27}$$

Görüldüğü gibi bir özdeğer -1'e eşittir. A^1 ve A^2 matrisleri ise Lemma 4.1 anlamında P-matrisleridir ve özdeğerlerin gerçel kısımları pozitiftir.

Teorem 4.7 doğrudan doğruya ters işaretli küme kalıplarının yakınsak olduğunu tanıtlamasa da bunu güçlü olarak ifade eder. Fakat Teorem 3.1 ve Teorem 4.7'nin ışığında aşağıdaki ifade doğrudur.

Ters işaretli küme kalıbı olan bir devre asla β bölgesi dışında yakınsayamaz. β bölgesinden başlayan veya geçen her yörünge, bu bölgede bir denge noktasına yakınsar. Ayrıca, β bölgesi dışındaki yörüngeler de bu bölgeye yönelirler. Yukarıdaki ifadeler deneysel olarak kanıtlanmıştır.

4.5 Denge Durumlarının Bulunması İçin Bir Algoritma

Hücrel nöral ağların denge noktalarını bulmak için L. Vandenberghe ve J. Vandewalle tarafından bir algoritma kullanılmıştır [22]. Bu algoritma parça lineer diferansiyel denklem takımlarını çözmek için geliştirilmiş algoritmaların, birden fazla denge durumunu hesaplayabilecek şekilde yeniden düzenlenmesinden elde edilmiştir. Problemin tanımı, hücrel ağların

$$T. \frac{dx}{dt} = -x + A.f(x) + B.u + I \quad (4.28)$$

şeklindeki dinamik denklemlerinin denge noktalarının bulunmasıdır. $f(\cdot)$, tanımı şekil 3.3 de verilen parça sürekli nöron çıkış fonksiyonudur. Denge noktaları türevin sıfıra eşit olduğu noktalardır ve çözüm

$$0 = -x + A.f(x) + B.u + I \quad (4.29)$$

$$x = A.f(x) + B.u + I \quad (4.30)$$

denkleminin çözümüne dönüşür. İfadenin analizini kolaylaştırmak için

$$A = I - D \quad -C = B.u + I \quad y = f(x) \quad (4.31)$$

dönüşümlerini yaparsak, denklemlerimiz

$$x = y - D.y - c \quad (4.32)$$

şekline dönüşür. y vektörü $C = \{ y \in \mathbb{R}^n \mid -1 \leq y_k \leq 1 \}$ kübü ile sınırlıdır. $y = f(x)$ 'in parça-sürekli yapısına göre (4.32) ifadesini irdelersek,

$$y_k = 1 \Rightarrow x_k = 1 - D.y_k^* - c \quad x_k - 1 = -(D.y_k^* + c) \geq 0 \\ (D.y_k^* + c) \leq 0$$

$$y_k = -1 \Rightarrow x_k + 1 = -(D.y_k^* + c) \leq 0 \\ (D.y_k^* + c) \geq 0 \quad (4.33)$$

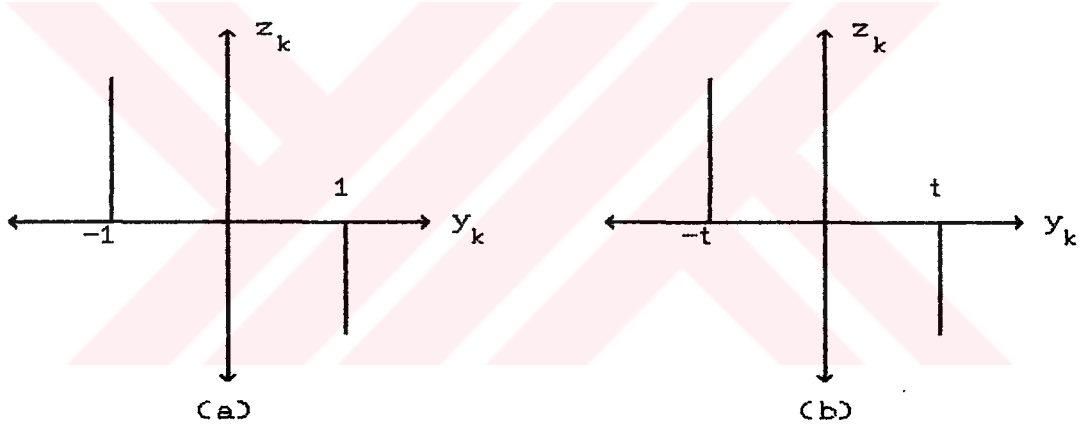
$$-1 \leq y_k \leq 1 \Rightarrow x_k = x_k - (D.y_k^* + c) \\ (D.y_k^* + c) = 0$$

sonucunu elde ederiz. Buradan denge noktası için şu ifadeyi verebiliriz.

$y^* \in C$ bir denge noktası olması için yalnız ve yalnız aşağıdaki koşulların sağlanması gereklidir:

$$\begin{aligned} y_k^* = 1 &\Rightarrow (D.y_k^* + c) \leq 0 \\ -1 < y_k^* < 1 &\Rightarrow (D.y_k^* + c) = 0 \\ y_k^* = -1 &\Rightarrow (D.y_k^* + c) \geq 0 \end{aligned} \quad (4.34)$$

(4.34) denklemleri, şekil 4.1a'da karakteristiği verilen $z = D.y + c$ türünden bir diyot-direnç devresini tanımlamaktadır. (4.34)'de verilen problem şekil 4.1b'de görül-



Şekil 4.1 (a) $z = D.y + c$ karakteristiği
(b) Düzenlenmiş karakteristik.

düğü gibi, $0 \leq t \leq \infty$ parametresi eklenerek genişletilmiştir. Algoritma $t=0$, $y=0$ noktasında başlar ve t , tek boyutlu parça-lineer bir eğriyi izleyerek değiştirilir. İzlenen yörüngenin C kübünün bir sınırını kestiği noktalar bir çözüme karşı düşer. Yörünge iki konumda olabilir, 1) sonsuza gider; 2) $D.x + c = 0$ 'ın bir çözümünde takılır. İkinci durumda, eğer sonuç noktası C kübünün içinde yer alıyorsa, bu (4.34) probleminin bir çözümüdür. Birinci durumda ise D 'nin tekil olmaması koşulu altında algoritma devam ettirilebilir.

[26] numaralı kaynakta algoritmanın hangi koşullar altında problemin çözümlerini bulabileceği ve bulunan çözümün tek olup, olmadığına ait teoremler de verilmiştir. Problemin veriliş şeklinden de anlaşılacağı üzere, algoritma sadece parça-lineer denklem takımlarının çözümüne yöneliktir ve hücresele ağların özelliklerine dayanmamaktadır. A matrisi içinde küme kalıbı ve komşuluk yapısının getirdiği özellikler kullanılmamıştır. Bu özelliklerin kullanılmasıyla çok daha efektif algoritmalar elde edilebilir.



BÖLÜM V

HÜCRESEL NÖRAL AĞLARIN UYGULAMALARI

HücreSEL nöral ağların ilk ve en büyük uygulama alanı görüntü işlemedir. Günümüzde işaret işleminin diğer alanlarında da hücreSEL ağlar ile çalışmalar yapılmaktadır. Literatürde görüntü işleme ve görüntüye dayalı algılama uygulamaları daha fazladır. HücreSEL ağlar, lineer-olmayan dönüşüm devreleri olmaları nedeniyle iki boyutlu görüntüler üzerinde özellik ayırtma, gürültü yok etme, hareket sezme gibi pek çok alanda kullanılmaktadırlar. Görüntü işlemede her piksel, hücreSEL ağın bir nöronuna karşı düşer. Görüntü genellikle hücreSEL ağın $x_{ij}(0)$ başlangıç konumuna $(-1,1)$ sürekli aralığında gerilim seviyeleri ile verilir. Nöron girişlerine ise devrenin programlanmasına göre gerekli kontrol akımları verilir veya verilmez. Ağ, dinamik yapısını belirleyen küme kalıpları ile çalışıp, bir denge durumuna yakınsadığında sonuç görüntü durumlarından veya nöron çıkışından elde edilir. Devre girişlerine ve durumlarına hangi değerlerin verileceği, sonucun nerede elde edileceği tamamen tasarımcının görüşüne bırakılmıştır ve bu konuda çalışmalar olmasına rağmen halen kesin bir yöntem bulunamamıştır.

HücreSEL ağlarda her kararlı sistem denge durumu iki boyutlu bir görüntüye eşdeğerdir. Kararlı sistem denge durumlarının sınırlı sayıda olduğunu bildiğimize göre, hücreSEL ağların sürekli $(-1,1)^{M \times N}$ görüntü uzayı ile sınırlı sayıda bipolar görüntüden oluşan bir sonuç uzayı arasında lineer-olmayan bir dönüşüm gerçekleştirdiğini söyleyebiliriz. Dönüşüm dinamik olduğundan sonucun varlığı kadar geçişin şekli de önemlidir. Geçiş

durumu devreyi başlangıç durumundan kararlı bir denge noktasına götüren yörüngedir. Her kararlı denge durumu ise bir grup yörüngenin limit noktasında bulunur. Bu özelliklerinden dolayı kararlı sistem denge durumlarına etki havuzları denmektedir. Bir küme kalıbı ve giriş akımları ile programlanmış hücrenel bir ağda tüm kararlı denge durumlarını bulma konusunda çalışmalar yapılmasına karşın, tüm noktaları bulmamızı sağlayacak bir algoritma geliştirilememiştir. Hücrenel ağlar asosiyatif bellek olarak kullanıldığında, kararlı denge durumları bellekte saklanan verilere karşı düşer. Hata düzeltici kodlarla çalışıldığında ise her kararlı denge noktası doğru bir koddur ve yörüngeler bir grup hatalı kodu alıp bu doğru noktaya yakınsatır.

Hücrenel ağların silikon üzerinde gerçekleşmesi konusunda çeşitli çalışmalar varsa da [27,28], bunlar henüz laboratuvar aşamasındadır. Araştırmaların çoğu bilgisayarlar da hazırlanmış devre simülatörleri veya özel uygulama programları ile yapılmaktadır. Devre analizi yöntemleri kullanıldığında, nöron yapısı içinde parça-lineer karakteristiğe sahip bir eleman bulunduğundan, bilinen analiz programları yerine PWLSPICE gibi parça-lineer devre elemanlarını da kullanabilen bir program gerekmektedir. Bu yöntem hayli zor olduğundan çoğunlukla hücrenel ağlar için geliştirilmiş komple simülasyon yapan özel programlar kullanılmaktadır. Bu tezde Macaristan Teknik Üniversitesi'nde Prof. T. Roska ve öğrencileri tarafından geliştirilen CNN Hücrenel Ağ Simülatörü (Bkz. Ek A) kullanılmıştır. CNN özel bir dosya formatında verilen durum, giriş ve küme kalıbı dosyalarını okuyarak devreyi çeşitli şekillerde çalıştırabilen ve sonuçları üreten bir programdır. CNN'de sürekli gerilimlerden oluşmuş giriş ve durum uzaylarını grafik olarak gösterebilmek için bir renk kodlaması kullanılmıştır. (-1,1) sürekli aralığına 8 adet renk şu şekilde karşı düşürülmüştür :

$[-1, -0.75]$	Beyaz	$[0, 0.25]$	Pembe
$[-0.75, -0.5]$	Açık Mavi	$[0.25, 0.5]$	Kırmızı
$[-0.5, -0.25]$	Yeşil	$[0.5, 0.75]$	Koyu Mavi
$[-0.25, 0]$	Sarı	$[0.75, 1]$	Siyah

Hücresel ağ simülasyonlarında giriş görüntüleri sayısal tarayıcılardan alınarak veya basit şekiller için elle hazırlanarak, CNN özel formatına dönüştürülür. Hazırlanan görüntü küme kalıpları ile ister adım adım her iterasyonda durarak, isterse de kararlı bir denge durumuna yakınsayana kadar sürekli olarak çalıştırılır ve eğer küme kalıbı kararlı bir duruma yakınsıyorsa sonuçlar gözlenir. Bazı küme kalıplarının nasıl çalıştığı, bulunmalarından sonra tümevarım yöntemiyle tanıtılabilmisse de, bir küme kalıbının devreyi nasıl etkileyeceği halen aktif bir araştırma konusudur.

Günümüze değin bulunmuş olan küme kalıplarını belirli başlıklar altında toplayabiliriz. Görüntü işleme problemlerinden gürültü yoketme, özellik ayırıştırma ve görüntüye dayalı veri işleme bunlardan başlıcalarıdır.

5.1 Gürültü Yoketme

Sayısal tarayıcılardan alınmış görüntülerde kağıt ve boya etkileri, kırışma, eskime gibi nedenlerle, bir iletişim ortamında transfer edilen görüntülerde de gürültüler nedeniyle bozulmalar meydana gelebilmektedir. Görüntü işleyicilerin hemen hepsinde, işlemlere başlamadan önce bu tür bozulmaları ortadan kaldıran gürültü yokedici devre parçaları kullanılır. Gürültüleri yoketme bir anlamda pikseller arasındaki ani renk farklarını ortadan kaldırmak yani en yakın komşularıyla bir ortalamasını almaktır. Doğal olarak gürültünün şekli devrenin çalışmasını büyük ölçüde etkiler. Hücresel nöral ağlarda gürültü yok etmek için iki farklı küme kalıbı önerebiliriz. Bunlardan ilki Gauss tipi gürültülerle bozulmuş

görüntüler için kullanılan

$$A = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 4 & 1 \\ \hline & 1 & \\ \hline \end{array}$$

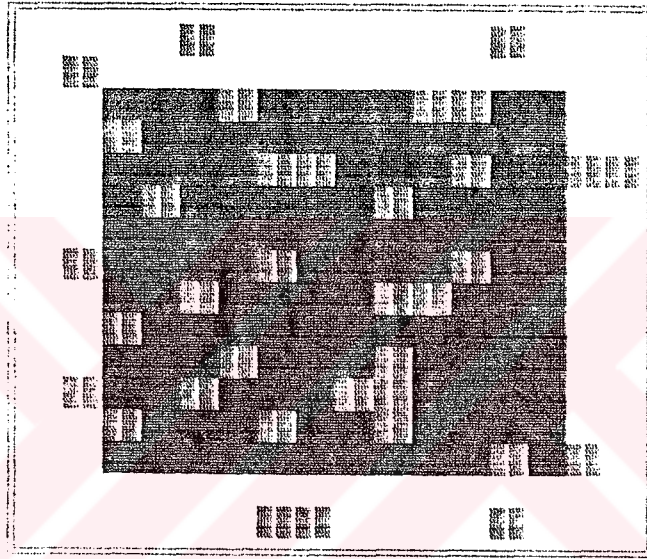
$$B=0$$

$$I=0$$

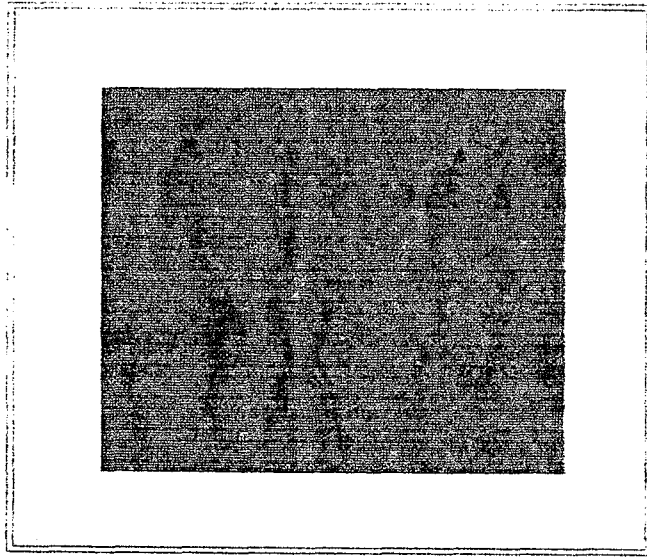
$$(5.1)$$

küme kalıbıdır. $\sigma=0.2$, $m=0$ parametreleriyle tanımlanmış Gauss gürültüsü ile bozulmuş olan bir karenin yukarıdaki küme kalıbı ile düzeltilişini Şekil 5.1'de gösterilmiştir.

a)



b)



Şekil 5.1 a) Gürültülü Şekil

b) Son Durum

Gerçek hayatta sıkça karşılaşıldığı gibi çoğu gürültü Gauss normlarına uymaz. Gauss tipinde olmayan gürültülerle bozulmuş görüntüler için aşağıdaki küme kalıbı da başarıyla kullanılabilir.

$$A = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 2 & 1 \\ \hline & 1 & \\ \hline \end{array} \quad B=0 \quad I=0 \quad (5.2)$$

5.2 Özellik Ayırıştırma Uygulamaları

Verilen bir şeklin içinden bir özelliğin çekilip çıkartılması, pek çok alanda karşımıza çıkan bir problemidir. Örneğin karakter tanıma algoritmalarında 44x44 pikselden oluşmuş bir karakteri incelemek yerine, bu karakterin belirli özelliklerini alıp, bu özellikleri daha önceden bildiğimiz karakterlerle karşılaştırmak, hız ve kolaylık açısından daha elverişlidir. Ayrıca bu tarz bir çalışma, tanıma algoritmasını son derece esnek yapacak, kişi, kalem, kağıt gibi dış etkenlerden doğrudan korunmamızı sağlayacaktır. Hücresel ağlarla şekillerin çeşitli özelliklerinin ayrıştırılması konusunda pek çok çalışma yapılmış ve büyük başarılar elde edilmiştir. Hücresel ağların bu konudaki en önemli avantajlarından biri, devrenin kararlı bir duruma yakınsama süresinin nöron sayısından bağımsız olması ve birkaç nöron gecikmesiyle sınırlı olmasıdır. Dolayısıyla şekil ne kadar büyürse büyüsün işlemin tamamlanma süresi aynı kalacaktır. Çeşitli özellik ayırıştırma kalıpları aşağıda incelenmiştir.

KENAR AYIRACI:

Hücresel ağlarda bir şeklin kenarlarını ayırştırmak için Laplace Operatörü de denilen aşağıdaki küme kalıbı

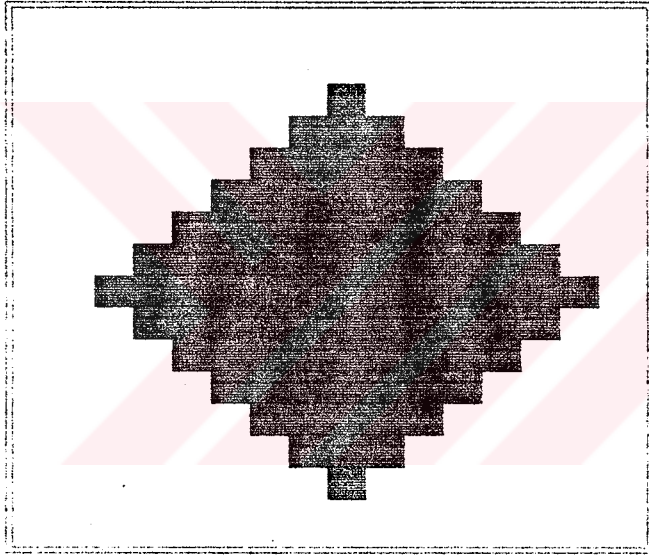
kullanılır :

$$A = \begin{bmatrix} & & \\ & 2 & \\ & & \end{bmatrix}$$

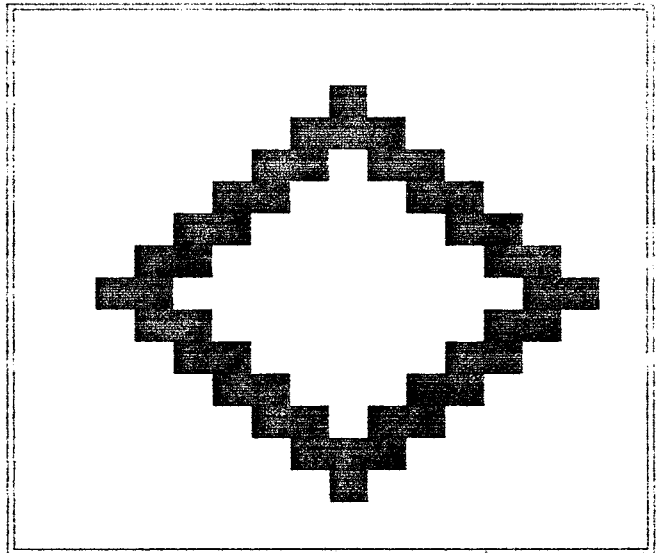
$$B = \begin{bmatrix} \frac{-1}{4} & \frac{-1}{4} & \frac{-1}{4} \\ \frac{-1}{4} & 2 & \frac{-1}{4} \\ \frac{-1}{4} & \frac{-1}{4} & \frac{-1}{4} \end{bmatrix}$$

$$I = -1.5 \quad (5.3)$$

Kenar ayırma küme kalıbının bir şekil üzerinde kullanılabilmesi için, devrenin girişlerine ve $v_{xij}(0)$ başlangıç durumuna şeklin verilmesi gereklidir. Yukarıdaki kalıbı kullanarak bir eşkenardörtgenin kenarlarının ayrıştırılması Şekil 5.2'de görülmektedir.



b)



Şekil 5.2 a) Orjinal Eşkenar Dörtgen
b) Kenar Ayırıcının Etkisi

KÖŞE AYIRACI:

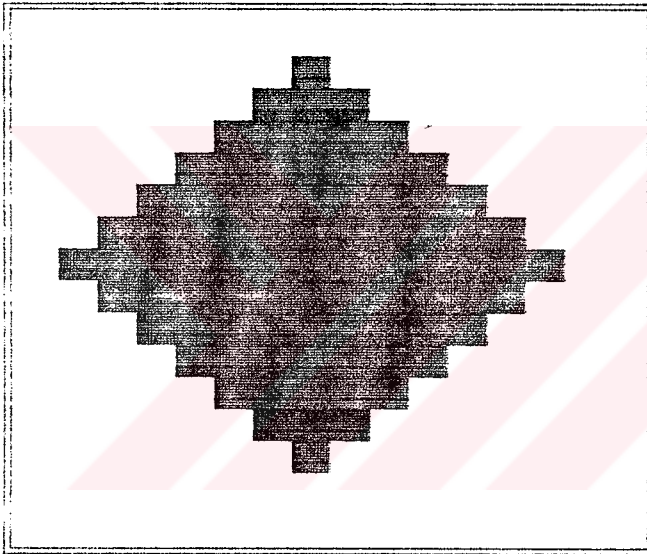
Yukarıda görülen kenar ayırıcı küme kalıbı üzerinde küçük bir değişiklik yaparak bir köşe ayıracını aşağıdaki gibi elde etmemiz mümkündür :

$$A = \begin{bmatrix} & & \\ & 2 & \\ & & \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{-1}{4} & \frac{-1}{4} & \frac{-1}{4} \\ \frac{-1}{4} & 2 & \frac{-1}{4} \\ \frac{-1}{4} & \frac{-1}{4} & \frac{-1}{4} \end{bmatrix}$$

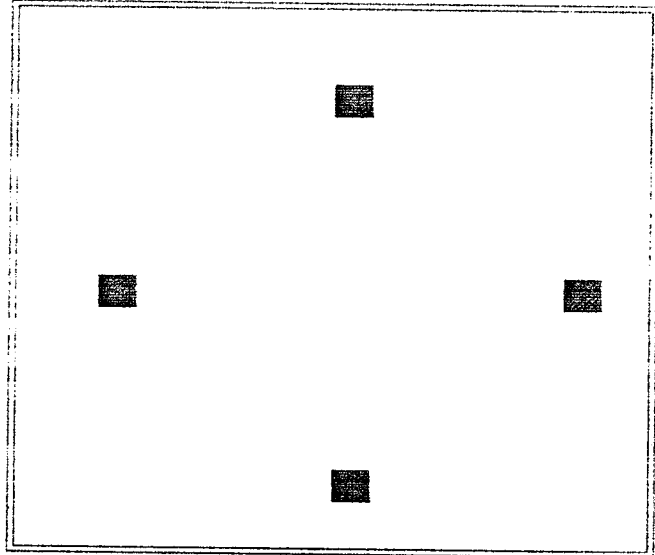
$$I = -3 \quad (5.3)$$

Köşe ayırıcı küme kalıbını gene eşkenardörtgen şeklimize uyguladığımızda Şekil 5.3'deki sonucu elde ederiz.



a)

b)



Şekil 5.3 a) Eşkenar Dörtgen

b) Köşe Ayırıcının Etkisi

BİRLEŞİK ALAN SAYACI:

Hücresele nöral ağların en önemli ve ilginç küme kalıplarından biri olan birleşik alan sayacı, Waseda Üniversitesi, Tokyo'da bulunmuştur [29]. Karakter tanıma sistemlerinde en çok kullanılan küme kalıbı olan birleşik alan sayacını şöyle verebiliriz :

$$A = \begin{bmatrix} -1 & 2 & 1 \end{bmatrix} \quad B=0 \quad I=0 \quad (5.4)$$

Oldukça basit olan bu küme kalıbı sadece tek doğrultuda sıfırdan farklı katsayılarla sahip olduğundan satır, sütun ve köşegen üzerinde çalıştırılabilir. Temel işlevi, görüntü içerisinde kaç tane kapalı alan var ise sonuçta o kadar sayıda tek piksel üretmektir. Birleşik alan sayacı bir anlamda analog öteleme saklayıcısına benzer. Örneğin Şekil 5.4 a)'da görülen satır, bu küme kalıbı ile çalıştırıldığında Şekil 5.4 b)'deki sonuç elde edilir. Asıl şekilde 4 adet kapalı siyah alan vardır. Sonuç şeklin sağında da 4 adet siyah piksel görülmektedir.



a)

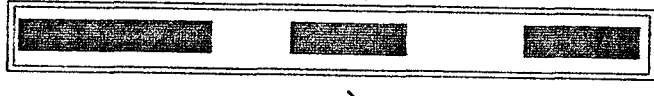


b)

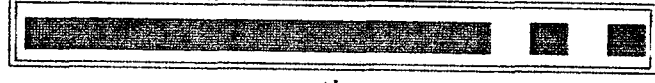
Şekil 5.4 a) Orjinal Şekil

b) BAS Etkisi

Önemli bir nokta hangi rengin fon, hangi rengin alan olarak algılanılacağına karar verilmesidir. Birleşik alan sayacı orjinal görüntünün en solundaki piksel rengini fon olarak kabul eder. $v_{x1}(0)=-1$ ise fon beyaz olarak seçilir ve siyah bölgeler sayılır. Aksi durumda $v_{x1}(0)=+1$ ise fon siyah olur ve beyaz kapalı bölgeler sayılır (Şekil 5.5).



a)



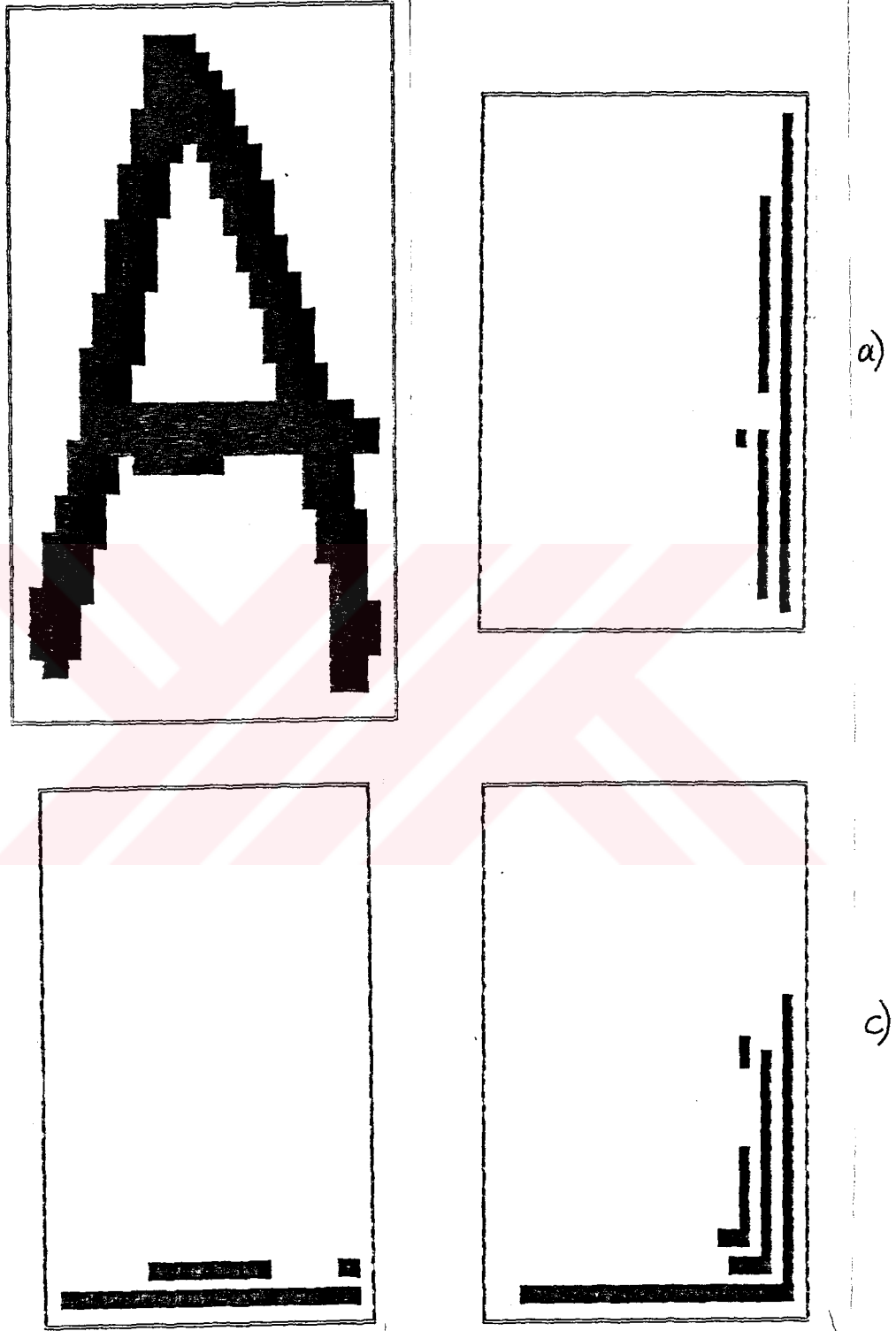
b)

Şekil 5.5 a) Siyah Fonlu Şekil

b) BAS Etkisi

Birleşik alan sayacı tek doğrultuda çalıştığından iki boyutlu bir görüntüye yatayda, düşeyde ve köşegende uygulanabilir. El yazması bir A harfinin bu 3 doğrultuda birleşik alan sayacı küme kalıbına uygulanmasıyla Şekil 5.6'daki sonuçlar elde edilmiştir.

Birleşik alanlar el yazması bir harf veya karakterde kişiye bağlı olarak değişmeyen pek az özellikten biri olduğu için, birleşik alan sayacı karakter tanıma sistemlerinde çok efektif olarak kullanılmaktadır. Her karakterin yatay, düşey ve çapraz eksenler üzerindeki birleşik alanlarının sayılması ve sonuçların bir tanıma algoritmasına verilmesiyle gerçekleştirilen sistemlerde %95-%98 düzeyinde başarı elde edilmiştir. Özellikle Japon alfabesi gibi çok fazla sayıda ve çok değişik şekillerde harflerden oluşan alfabeleri tanımakta birleşik alan sayacı önemli rol oynar. Çünkü tanımanın zor olması yanında algılanan görüntünün daha önceden bilinen görüntülerle karşılaştırılması ve en yakın olanına eşleştirilmesi gereklidir. Bu küme kalıbı bize, görüntüleri oldukları gibi örneğin 44x44 matrisler yerine, eksenlerindeki kapalı alan sayılarıyla saklama imkanı verir ve önemli ölçüde veri sıkıştırması yaparak, bellekten tasarruf sağlar. Waseda Üniversitesinde bu tarzda çalışan bir karakter tanıma sistemi gerçekleştirilmiştir.



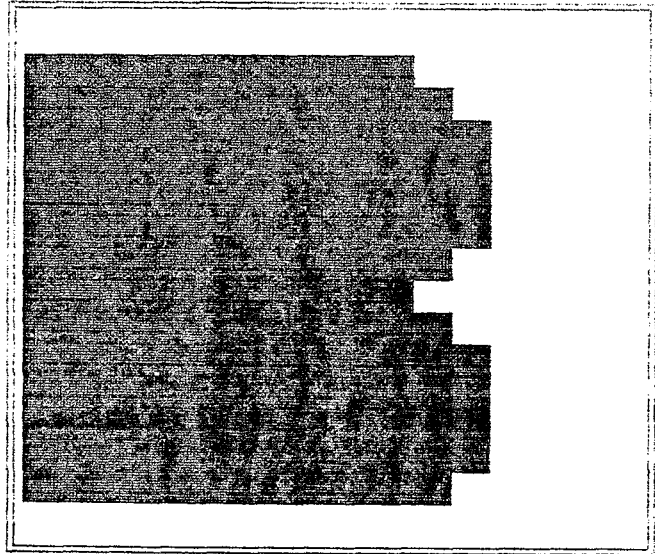
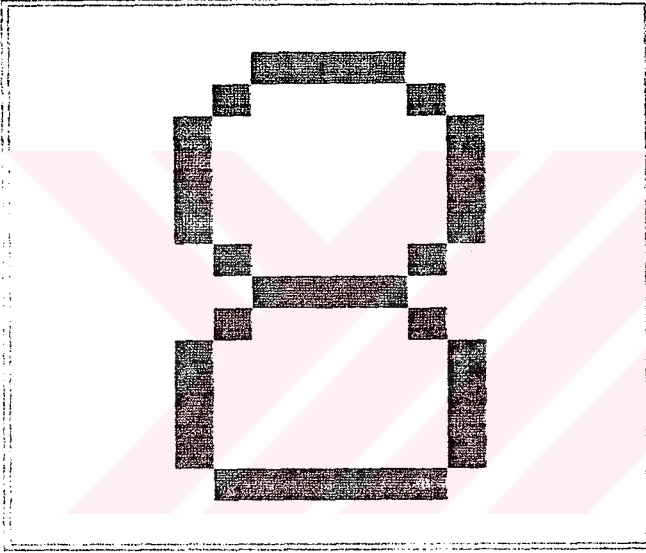
Şekil 5.6 El Yazması A harfine BAS Uygulanması
a) Yatay Tarama b) Düşey Tarama
c) Çapraz Tarama

GÖLGE YARATMA:

Waseda Üniversitesi, Tokyo'da bulunan bu küme kalıbı, görüntülerin sağdan bir ışık kaynağı ile aydınlatılmaları durumunda alacakları gölgeyi vermektedir. Küme kalıbı

$$A = \begin{bmatrix} 0 & 2 & 2 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 2 & 0 \end{bmatrix} \quad I = 0 \quad (5.5)$$

olarak bulunmuştur. Bir karakterin gölgesini elde etmek için bütün başlangıç durumları +1 yapılır ve devrenin



Şekil 5.7 Gölge Yaratma Uygulaması

girişlerine orjinal karakter verilir. Şekil 5.7'de sekiz sayısının gölgesinin elde edilmesi yer almaktadır. Gölge yaratma küme kalıbının devre üzerindeki etkisinin matematiksel tanıtı [30] numaralı referansta bulunabilir. Tanıt tümervarım metoduyla ve tüm olası durumların incelenmesi ile yapılmıştır.

BOŞLUK DOLDURMA:

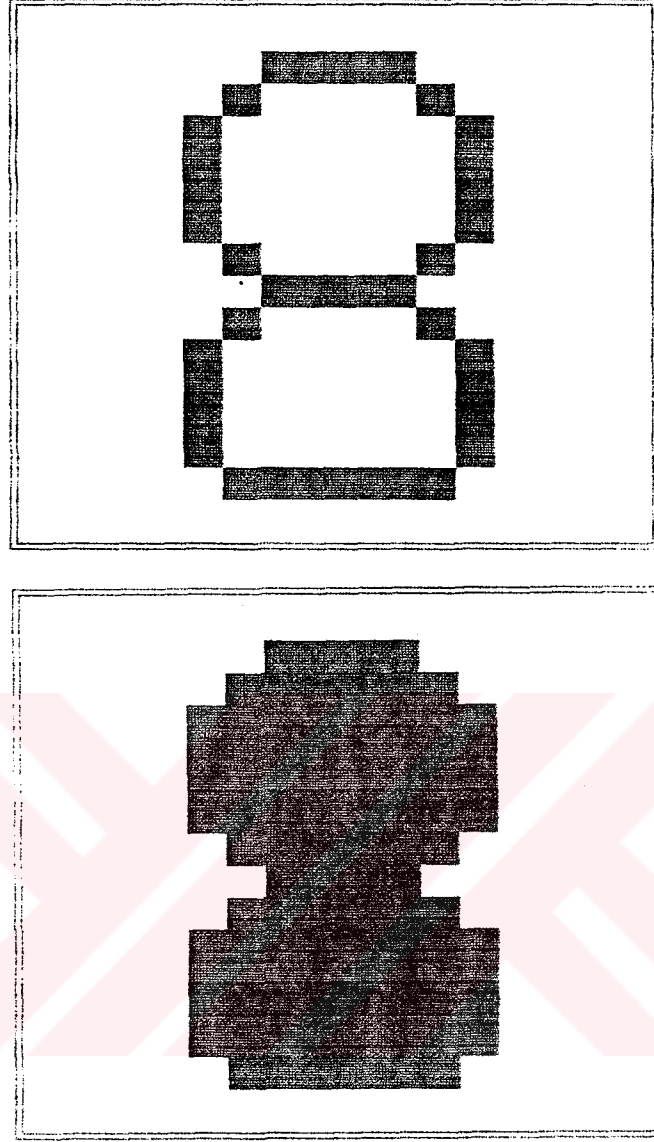
Waseda Üniversitesinde bulunan küme kalıplarından bir diğeri [31]. Boşluk doldurucu küme kalıbı, verilen bir şeklin içindeki siyah piksellerle çevrelenmiş kapalı beyaz alanları siyahla doldurmaktadır. Küme kalıbı

$$A = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 2 & 1 \\ \hline & 1 & \\ \hline \end{array} \quad B=4 \quad I=-1 \quad (5.6)$$

şeklinde verilmiştir. Devrenin çalışması için başlangıç durumlarının hepsine $x_{ij}(0)=+1$ değeri verilir. Orjinal görüntü ise nöron girişlerine uygulanır. Devre çalıştırılıp kararlı duruma yakınsadığında, görüntü içindeki kapalı boşluklar doldurulmuş olur. Şekil 5.8'de gene sekiz sayısının boşluk doldurma küme kalıbı uygulandıktan sonraki hali verilmiştir. Boşluk doldurma küme kalıbının devre üzerindeki etkisinin matematiksel tanıtı [31] numaralı referansta yer almaktadır.

GÖRÜNTÜ TÜMLEME :

Hücresel ağlarla bipolar bir görüntünün tümleyenini bulmak da mümkündür. Giriş görüntüsü $[+1, -1]$ değerlerinden oluştuğunda, tümleme kalıbı görüntü içindeki tüm piksellerin evriğini alır. Bu işlem daha çok ikili hesap uygulamalarında kullanılmaktadır. Görüntüler üzerinde lojik işlemlerin, küme kalıpları ile de yapılabileceğini göstermesi açısından tümleme kalıbı önemli bir kalıptır.



Şekil 5.8 Boşluk Doldurma Uygulaması

Tümleme kalıbını şöyle verebiliriz :

A=

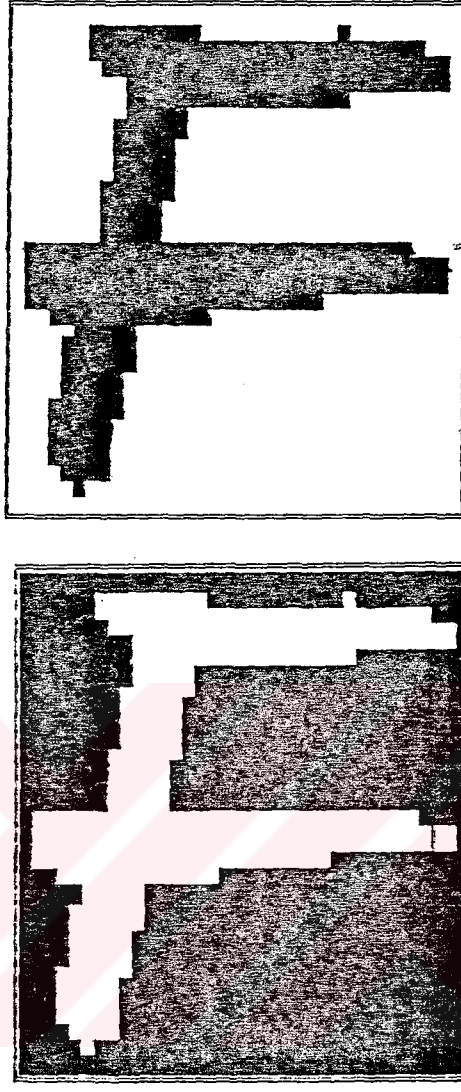
	1	

B=

	-2	

$$I=0 \quad (5.7)$$

Giriş görüntüsü aynı anda hem $v_{xij}(0)$ başlangıç durumuna hem de devrenin girişlerine verilir. Devre kararlı bir duruma yakınsadıktan sonra, çıkışlarda esas görüntünün tümleyeni elde edilir (Şekil 5.9).



Şekil 5.9 Tümleme Uygulaması

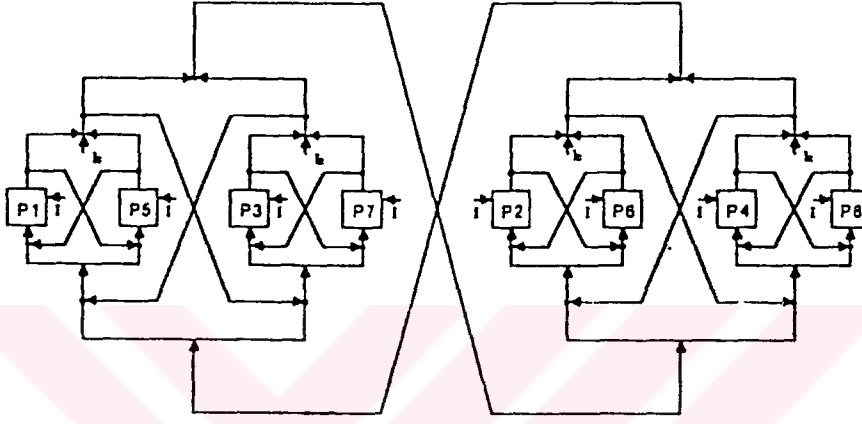
GÖRÜNTÜ İNCELTME :

Görüntü inceltme, bir şeklin bağlantı özelliğini bozmadan, bir piksel kalınlığında ikili $[0,1]$ veya bipo- lar $[-1,1]$ başka bir şekle dönüştürülmesidir. Özellikle el ile yazılan karakterlerin kalınlık farklarından kurtarılmasında kullanılır. Görüntü inceltmede iki temel iş- lev bulunmaktadır :

- 1) Şeklin kalınlığının azaltılması,
- 2) Kalınlık bir piksele ulaştığında inceltme işleminin durdurulması.

İkinci işlem birinciye göre daha zordur ve dikkat ister.

Çünkü inceltme işlemi tam zamanında durdurulmazsa şekil tamamen yokolabilir veya bağlantı yapısı bozulabilir [32,33]. Hücresel ağlarda görüntü inceltme işlemi için tek bir küme kalıbı değil, 8 gruplu değişik bir mimari kullanılmıştır. $v_{xp} \in \mathbb{R}^{M \times N}$, $p=1,2,...,8$ olmak üzere sekiz gruba ait durum vektörleri olsun. Gruplar arası bağlantı Şekil 5.10'da görüldüğü gibidir.



Şekil 5.10 Görüntü Inceltme Devresinin Yapısı

Hücresel ağın dinamik denklemi $R=1K\Omega$, $C=1nF$, $I=I_e=-1.9mA$ olmak üzere,

$$C \frac{dv_{xp}}{dt} = \frac{-1}{R} v_{xp} + A_{pp} * v_{yp} + A_{pq} * v_{yq} + I + k \sum_{\substack{r \neq p+4 \\ r \neq p}} (v_{yr} + I_e) \quad (5.8)$$

$$q=(p+4) \bmod 8$$

$$I=(I, I, \dots, I)$$

$$I_e=(I_e, I_e, \dots, I_e) \in \mathbb{R}^{M \times N}$$

$$v_{yq}=(v_{yq11}, v_{yq12}, \dots, v_{yqMN})$$

$$v_{ypij} = \frac{1}{2} \left[|v_{xpij} + 1| - |v_{xpij} - 1| \right] \quad \begin{matrix} 1 \leq i \leq M \\ 1 \leq j \leq N \end{matrix}$$

Tablo 5.1 İnceltici Kalıplar

1.0	1.0	0.0
1.0	2.0	-1.0
0.0	-1.0	0.0

P No.1
A₁₁

1.0	1.0	1.0
0.0	3.0	0.0
-1.0	-1.0	-1.0

P No.2
A₂₂

0.0	1.0	1.0
-1.0	2.0	1.0
0.0	-1.0	0.0

P No.3
A₃₃

1.0	0.0	-1.0
1.0	3.0	-1.0
1.0	0.0	-1.0

P No.6
A₆₆

-1.0	0.0	1.0
-1.0	3.0	1.0
-1.0	0.0	1.0

P No.4
A₄₄

0.0	-1.0	0.0
1.0	2.0	-1.0
1.0	1.0	0.0

P No.7
A₇₇

-1.0	-1.0	-1.0
0.0	3.0	0.0
1.0	1.0	1.0

P No.8
A₈₈

0.0	-1.0	0.0
-1.0	2.0	1.0
0.0	1.0	1.0

P No.5
A₅₅

Tablo 5.2 Durdurucu Kalıplar

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	8.0	-1.0	0.0
0.0	0.0	-1.0	-2.0	0.0
0.0	0.0	0.0	0.0	0.0

A₁₅(i,j,k,l)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	8.0	0.0	0.0
0.0	-1.0	-2.0	-1.0	0.0
0.0	0.0	0.0	0.0	0.0

A₂₆(i,j,k,l)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	-1.0	8.0	0.0	0.0
0.0	-2.0	-1.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

A₃₇(i,j,k,l)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	-1.0	0.0
0.0	0.0	8.0	-1.0	-1.0
0.0	0.0	0.0	-1.0	0.0
0.0	0.0	0.0	0.0	0.0

A₄₆(i,j,k,l)

0.0	0.0	0.0	0.0	0.0
0.0	-1.0	0.0	0.0	0.0
0.0	-2.0	8.0	0.0	0.0
0.0	-1.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

A₄₈(i,j,k,l)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	-2.0	0.0	0.0
0.0	0.0	8.0	-2.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

A₇₇(i,j,k,l)

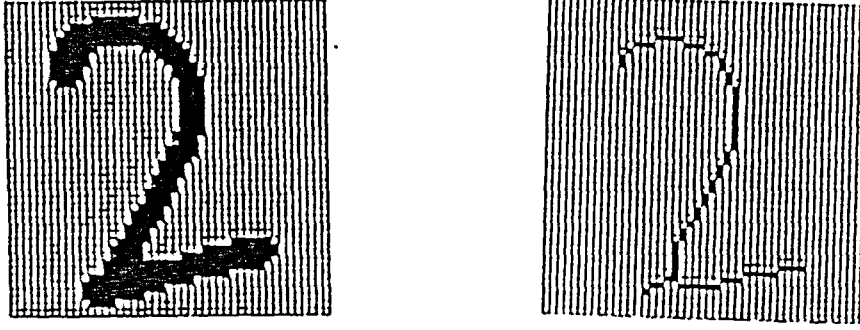
0.0	0.0	-1.0	0.0	0.0
0.0	-1.0	-1.0	-1.0	0.0
0.0	0.0	8.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

A₈₂(i,j,k,l)

0.0	0.0	0.0	0.0	0.0
0.0	0.0	-2.0	0.0	0.0
0.0	-2.0	8.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

A₃₁(i,j,k,l)

Görüntü inceltme dinamik denklemlerinde kullanılan küme kalıpları Tablo 5.1 ve Tablo 5.2'de görülmektedir. Görüntü inceltme işlemine bir örnek de Şekil 5.11'de verilmiştir.



Şekil 5.11 Görüntü inceltme uygulaması

İKİLİ HESAP YÖNTEMİYLE GÖRÜNTÜ INCELTME :

T.Roska ve G.Bartfai tarafından görüntü inceltme problemine farklı bir çözüm getirilmiştir [34]. Matsumoto ve arkadaşlarının çok katmanlı, karmaşık görüntü inceltme sistemine karşılık, Roska ve Bartfai, temelde basit fakat ardışıl olarak tekrarlandığında aynı sonucu veren bir küme kalıbı ve hesaplama yöntemi bulmuştur. Bu yöntem işlemlerin ardışıl olarak tekrarlanmasını gerektirdiğinden, diğer yöntemle oranla daha uzun süre olsa da, günümüz yarıiletken teknolojisi düşünüldüğünde yakınsama süresi gene mikrosaniyeler mertebesinde olacaktır.

Roska bu çalışmasında hücresel ağ teorisi ile klasik lojik teoriyi birarada kullanmıştır. Görüntü inceltme yönteminde iki adet küme kalıbı kullanılır. Bunlardan ilki

$$A = \begin{bmatrix} & & \\ & 2 & \\ & & \end{bmatrix} \quad B = \begin{bmatrix} -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & 2 & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \end{bmatrix} \quad I = -1.5 \quad (5.9)$$

şeklindeki İnceltme Kalıbı, diğeri ise

$$A = \begin{array}{|c|c|c|} \hline & & \\ \hline & 2 & \\ \hline & & \\ \hline \end{array}$$
$$B = \begin{array}{|c|c|c|} \hline \frac{-1}{2} & \frac{-1}{2} & \frac{-1}{2} \\ \hline \frac{-1}{2} & -6 & \frac{-1}{2} \\ \hline \frac{-1}{2} & \frac{-1}{2} & \frac{-1}{2} \\ \hline \end{array}$$

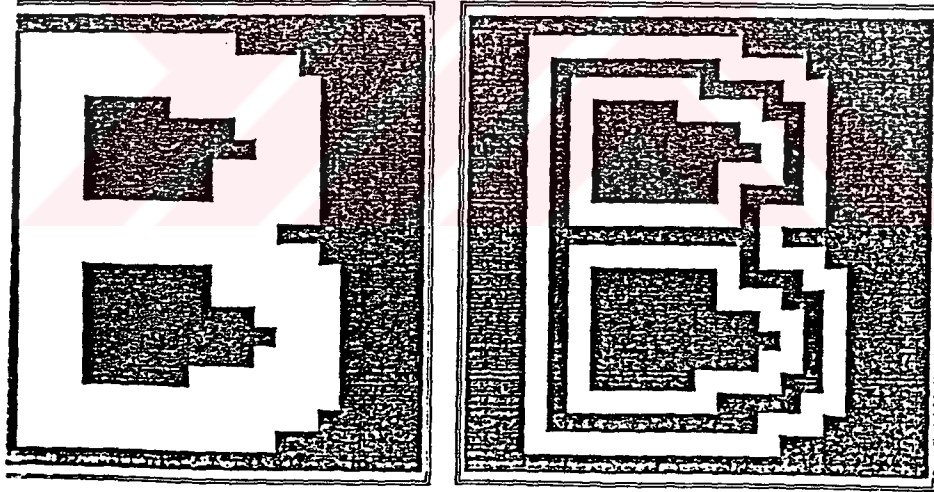
$$I = -1.5 \quad (5.10)$$

şeklindeki Tümleme Kalıbıdır. Bir görüntüyü inceltmek için şu yöntem izlenir :

Adım 1: Tümleme kalıbı ile şeklin değili alınır.

Adım 2: 1. adımda elde edilen şekle inceltme kalıbı uygulanır.

Adım 3: 2. adımda elde edilen şekil ile orjinal şeklin herbir pikseli karşılıklı olarak VE işlemine sokulur. Sonuçta elde edilen şekil bir piksel kalınlığında ise işlem sonlanır, değilse adım 1'e dönülerek işlemler tekrarlanır. Analog algoritmanın adımları Şekil 5.12'de görülebilir.



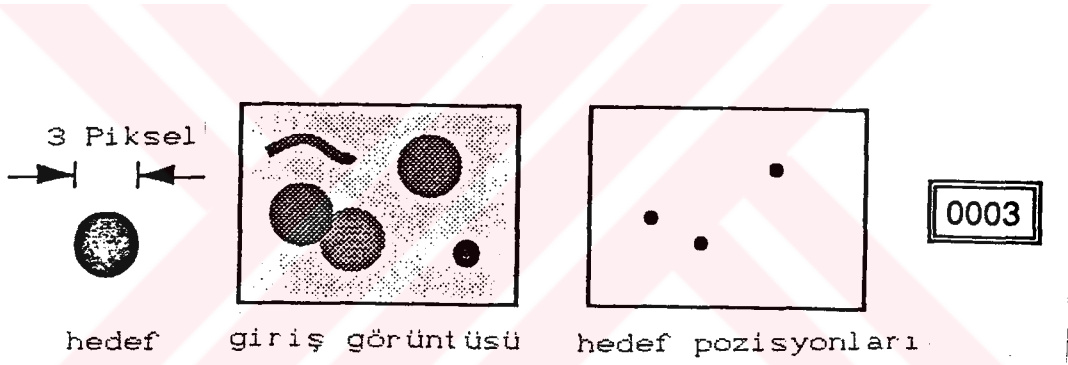
Şekil 5.12 İkili Hesap Yöntemiyle Görüntü İnceltme

5.3 Hücresel Ağlarla Küçük Cisim Sayma

Bir görüntü içindeki belirli cisimlerin sayılması, pek çok alanda ihtiyaç duyulan bir işlemdir. Örneğin kan testlerinde alyuvarların sayılması gerekirken farklı

aplarda olan diğ er h  creciklerin ayırdedilmesi gerekmektedir. Astronomik alıřmalar sırasında da yıldız k  mele-
rinin sınıflanması, yerlerinin belirlenmesi gibi durum-
larda da bu t  r sayım iřlemleri kullanılır. Tanım ola-
rak cisim sayma, bir g  r  nt   iinde řekli, boyutları ve
konuřlanıřı bilinen bir hedeften ka adet olduėunun be-
lirlenmesidir. Sayma iřleminin en zor yanı hedef cisim-
lerin birbirlerinin   zt  nde akıřık olarak da bulunabil-
meleridir.

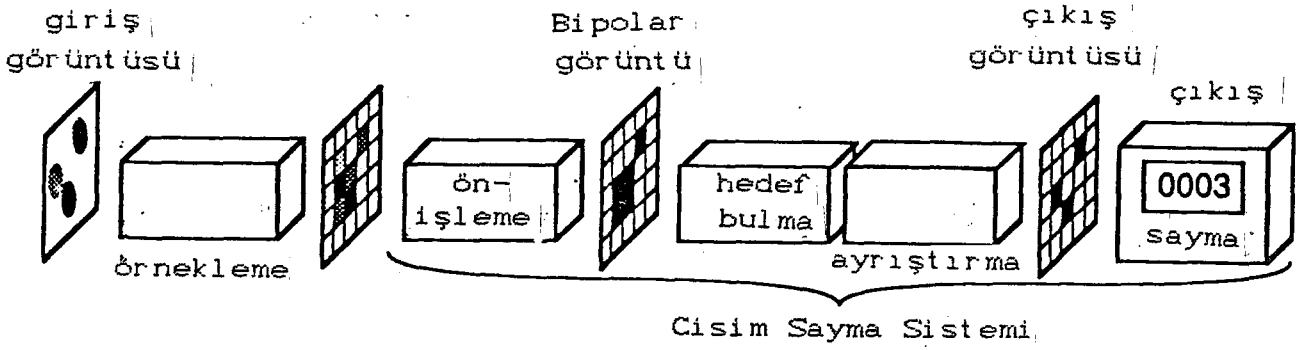
Sayma iřlemine basit bir  rnek řekil 5.13'de veril-
miřtir. apı g  r  nt  n  n   z  n  rl  ė ne g  re 3 piksel bo-
yutunda olan siyah bir disk, hedef olarak seilmiřtir.
Giriř g  r  nt  s  nde ise fon ile diskler arasındaki ton
farkı azdır ve sabit deėildir. ıkıř g  r  nt  s  nde, disk-



řekil 5.13 Cisim Sayma  rneėi

lerin merkezlerinde birer siyah nokta g  r  lmektedir. Siyah noktaların sayılmasıyla giriř g  r  nt  s  nde 3 adet disk olduėu bulunur.

G  r  nt   iinde cisim sayma iřlemi řekil 5.14'deki adımlardan oluřur. G  r  nt     nce kabul edilebilir bir   z  n  rl  kte, ayrıık piksel deėerlerine d  n  řt  r  l  r, daha sonra da bipolar g  r  nt   haline getirilir. Bipolar d  n  ř  m esnasında g  r  nt  y   g  r  lt  lerden arındırmak iin devrenin bařlangı durumlarına g  r  nt   verilip ařaėıdaki k  me kalıbı devreye uygulanır.



Şekil 5.14 Cisim Sayma Sistemi

$$A = \begin{array}{ccccc} & & 1 & & \\ & 1 & 6 & 1 & \\ & & 1 & & \end{array} \quad B=0 \quad I=0 \quad (5.11)$$

Görüntü bu aşamadan sonra ayrıştırma işlemi için hazırdır. Öncelikle hedef cisim, verilen görüntü içinde uygun küme kalıbı ile aranır. Seçtiğimiz 3 piksel genişliğindeki disk için aşağıdaki küme kalıbı kullanılır :

$$A=2 \quad I=-8 \quad B = \begin{array}{ccccc} & & -\frac{1}{2} & & \\ & -\frac{1}{2} & 1 & 1 & -\frac{1}{2} \\ & -\frac{1}{2} & 1 & 1 & -\frac{1}{2} \\ & -\frac{1}{2} & 1 & 1 & -\frac{1}{2} \\ & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \end{array} \quad (5.12)$$

$v_{xij}(0)=0$

Küme kalıbını çalıştırmak için devrenin bütün başlangıç koşulları sıfır yapılarak esas görüntü devrenin girişlerine verilir. Görüntü içinde hedefe benzeyen bölgeler

bulunduktan sonra yapılacak tek yapılacak iş üst üste gelen hedefler var ise bunların birbirinden ayrılmasıdır. Ayırma işlemi için görüntü devrenin başlangıç durumuna verilerek aşağıdaki küme kalıpları ardışıl olarak uygulanır.

$$A = \begin{array}{|c|c|c|} \hline \frac{-9}{10} & 2 & \frac{-11}{10} \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline & \frac{-9}{10} & \\ \hline \frac{-11}{10} & 2 & \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline \frac{-11}{10} & & \\ \hline & 2 & \\ \hline & \frac{-9}{10} & \\ \hline \end{array} \quad (5.13)$$

Yukarıda verilen işlemler ardışıl olarak giriş görüntüsüne uygulandığında, esas görüntü içindeki hedeflerin merkezlerini belirten siyah noktalardan oluşmuş bir sonuç elde edilir. Buradaki noktaların sayılmasıyla işlem tamamlanmış olur.

BÖLÜM VI

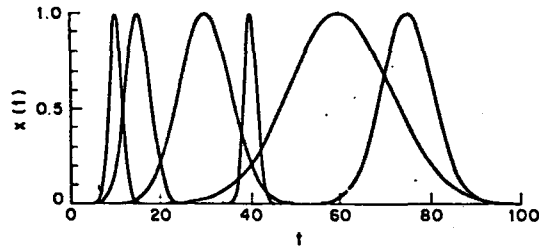
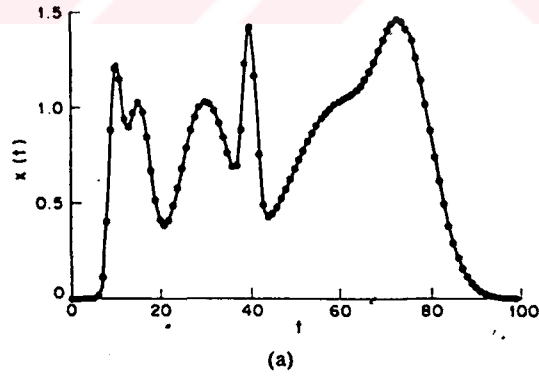
HNA YAPISININ İŞARET AYRIŞTIRMA PROBLEMİNE UYGULANMASI

Hücresel ağların genel yapısı ve özellikleri incelendikten sonra bu tezde, işaret işleminin önemli problemlerinden biri olan işaret ayrıştırma problemi hücre- sel nöral ağ yapısı kullanılarak çözülmüştür. Nöral ağ- lar işaret ayrıştırma problemine ilk defa 1986 yılında Hopfield ve Tank tarafından uygulanmıştır [37]. Bu ça- lışmada tam bağlı bir Hopfield Ağı ve ortogonal olmayan Gauss tipi baz fonksiyonları kullanılmıştı. Hopfield ve Tank'ın çalışmasından yola çıkarak bu tezde hücre- sel ağ yapısına dayanan daha esnek bir çözüm bulunmuştur. Hopfield ve Tank çözümündeki önemli dezavantajlardan biri, ayrıştırılacak her yeni analog işaret için devre parametrelerinin yeniden hesaplanması gereği idi. Oysa hücre- sel ağ çözümünde tek bir küme kalıbı ile her türlü işaret ayrıştırılabilmektedir. Ayrıca Hopfield ve Tank sadece pozitif katsayılı Gauss fonksiyonları kullandık- larından, ancak pozitif değerli fonksiyonlar üzerinde çalışabilmekteydiler. HNA çözümünde ise pozitif ve ne- gatif ayrıştırma küme kalıpları ile bipolar işaretler de işlenebilmektedir. HNA'da çözümün duyarlılığı, kullanılan devrenin satır sayısına bağlı olarak değişmektedir. Hücre- sel ağların yakınsama süresi ve hücre potansiyelle- rinin alacağı maksimum değer devrenin boyutlarından ba- gımsız olduğu için istenen duyarlılığı sağlayacak şekilde devrenin boyutları arttırılabilir.[38,39,40].

6.1 Problemin Tanımı ve Hopfield Çözümü

İşaret ayırıştırma problemi genel olarak şöyle tanımlanır: N noktada örneği alınan analog bir işaretin, bir grup baz fonksiyonunun lineer kombinasyonundan oluşmuş en yakın ifadesini bulmaktır. Nöral devrelerde bağlantı katsayıları genellikle hücre çıkışlarının çarpım terimlerinin yanında yer aldıklarından, baz fonksiyonlarının ortogonal olmaması tercih edilir. Yukarıdaki tanıma dayanarak işaret ayırıştırma probleminin bir optimizasyon problemi olduğunu söyleyebiliriz. Amaç orjinal analog değerler ile bizim bulduğumuz fonksiyonun değerleri arasındaki hatayı minimum yapmaktır. Hücresel ağlar da Liapunov fonksiyonlarını minimize ederek çalıştıklarından probleme doğal bir uygunluk gösterirler. Çözüme ulaşmak için ilk adım kullanılacak baz fonksiyonlarına karar verilmesidir. Hopfield ve Tank çalışmalarında sabit genlikli, değişken genişlikte Gauss fonksiyonlarını kullanmışlardır. Bu fonksiyonlar

$$\epsilon_{\sigma\mu}(t) = e^{-\left[\frac{t-\mu}{\sigma}\right]^2} \quad (6.1)$$



Şekil 6.1 Hopfield İşaret Ayırıştırması

şeklinde ifade edilirler. $E_{\sigma\mu}(t)$ fonksiyonlarıyla işaretleme ayrıştırmaya bir örnek Şekil 6.1'de verilmiştir. Hopfield çözümünde her nöron, σ genişlikli ve μ tepe noktası parametreleriyle ifade edilen bir Gauss fonksiyonuna karşı düşer. σ sınırlı sayıda elemandan oluşmuş bir kümeden değerler alırken, örneğin $\sigma_{\max}$ adet, μ örneklem noktaları olarak 1..N'e kadar değerler almaktadır. Her nöronun çıkışı, o nörona ait fonksiyonun gerçek fonksiyon içinde var olup olmadığını gösteren $V_{\sigma\mu}$ değeridir. $V_{\sigma\mu} + 1$ değerini alıyorsa o nörona ait Gauss fonksiyonu çözümde var, 0 değerini alıyorsa yok demektir. Ayrıştırma işlemi tamamlandığında minimum değerini alan Hopfield devresi enerji fonksiyonu şöyledir:

$$E = \frac{1}{2} \sum_{n=1}^N \left[x(n) - \sum_{\sigma=1}^{\sigma_{\max}} \sum_{\mu=1}^N V_{\sigma\mu} E_{\sigma\mu}(n) \right]^2 - \frac{1}{2} \sum_{n=1}^N \sum_{\sigma=1}^{\sigma_{\max}} \sum_{\mu=1}^N (E_{\sigma\mu}(n))^2 [V_{\sigma\mu}(V_{\sigma\mu}-1)] \quad (6.2)$$

Yukardaki enerji fonksiyonunu Hopfield Ağı enerji fonksiyonu

$$E = - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N T_{ij} V_i V_j - \sum_{i=1}^N V_i I_i \quad (6.3)$$

ile karşılaştırdığımızda T bağlantı ağırlıkları matrisi ve I giriş akımı şu şekilde bulunur :

$$T_{\sigma\mu, \sigma'\mu'} = \sum_{n=1}^N \frac{1}{e} \left[\left[\frac{n-\mu}{\sigma} \right]^2 + \left[\frac{n-\mu'}{\sigma'} \right]^2 \right] \quad (6.4)$$

$$I_{\sigma\mu} = \sum_{n=1}^N x(n) \frac{1}{e} \left[\left[\frac{n-\mu}{\sigma} \right]^2 \right] - \frac{1}{2} \sum_{n=1}^N \frac{1}{e} \left[\left[\frac{n-\mu}{\sigma} \right]^2 \right] \quad (6.5)$$

(6.5) ifadesinde görüldüğü gibi $x(n)$ örnek değerleri sabit akım kaynağı içinde yer almaktadır. Bu durumda her yeni örnek grubu için akım değerinin yeniden hesaplanması gereklidir. T matrisi hesaplandığında da Gauss fonksiyonlarının yapısı nedeniyle fazlasıyla ayrık bir matris ortaya çıkmaktadır. Hücresel ağların kullanılmasıyla bu dezavantajlar ortadan kaldırılmıştır.

6.2 Hücresel Ağlar ile Gerçekleme

İşaret ayrıştırma problemi hücresel ağlara uygulanırken Gauss fonksiyonlarının değişik bir türü baz fonksiyonu olarak kullanılmıştır:

$$\epsilon_{\mu}(t) = e^{-\frac{1}{2}\left[\frac{t-\mu}{\sigma}\right]^2} \quad (6.6)$$

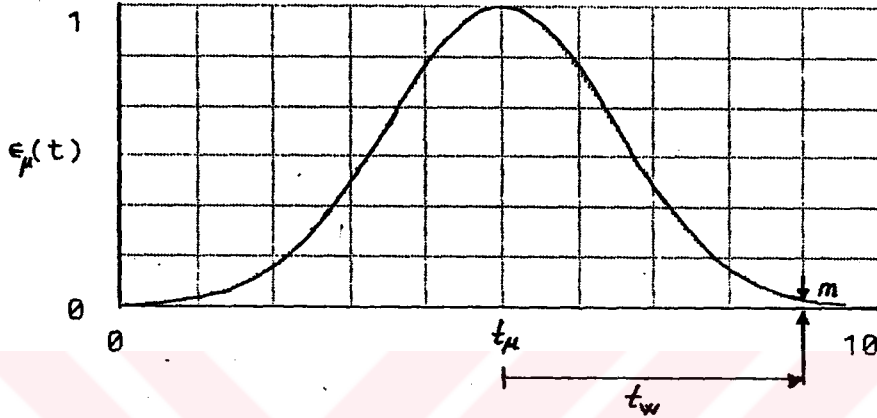
(6.6) fonksiyonlarının genlikleri 1, genişlik parametreleri σ ise sabittir. Hücresel ağımız $M \times N$ boyutlu olsun. Orjinal fonksiyon N noktada örneklenip devreye verildikten sonra hücresel ağ hangi noktalar bir Gauss fonksiyonu yerleştirebileceğini ve bu fonksiyonların genliklerini bulur. Genlikler kombinasyon katsayıları olmak üzere orjinal fonksiyonun yaklaşığı $f_{est}(t)$ (6.6) baz fonksiyonlarının lineer kombinasyonu olarak (6.7) ifadesinden elde edilir.

$$f_{est}(t) = \sum_{i=1}^N c_i \epsilon_i(t) \quad c_i = \sum_{j=1}^M V_{ji} \quad (6.7)$$

(6.7) ifadesinde V_{ij} (i, j) hücreesine ait Gauss fonksiyonunun orjinal fonksiyonda olup olmadığını gösteren değerdir ve $V_{ij}=1$ ise fonksiyon var, $V_{ij}=0$ ise fonksiyon yok anlamına gelir. Hücresel ağlarda ise hücre çıkışları bipolar değerler alan $\{+1, -1\}$ fonksiyonlardı. Kullanılan küme kalıbının pozitif veya negatif ayrıştırma kalıbı olmasına göre, hücre çıkışının sırasıyla +1 veya -1

olması, o fonksiyonun çözüm içindeki varlığını gösterir.

Baz fonksiyonlarının genişliğini belirleyen σ parametresi, gerçek fonksiyonun T örnekleme aralığına bağlı olarak bulunur. Şekil 6.2'de görüleceği gibi Gauss fonksiyonları tepe noktasından kısa bir süre sonra sifıra yaklaşmakta ve asimptotik olarak limitte sifıra ulaşmaktadır. Hesaplamalara kolaylık getirmesi için



Şekil 6.2 Gauss Karakteristiği

$e_{\mu}(t_v)=m=0.0001$ noktası altında fonksiyon sıfır olarak kabul edilmiştir. İşaret ayrıştırma problemi çözülürken hücrel ağın komşuluk yapısı, pratik gerçeklemeleri kolaylaştırmak için 1 olarak seçilir. 1-komşuluk seçilmesinin Gauss fonksiyonları açısından anlamı, bir örnek noktasındaki fonksiyonun ancak iki yanındaki örnek noktaları etkilemesi, daha sonra hemen sifıra düşmesidir. Bu nedenle tepe noktasından iki örnekleme aralığı sonra fonksiyon sifıra inmiş olmalıdır. Matematiksel olarak durumu şöyle ifade edebiliriz:

$$t_v = 2T \quad (6.8a)$$

$$t_v = \sqrt{-2.\sigma^2 \ln \frac{m}{c}} \quad (6.8b)$$

(6.8) ifadelerinden orjinal fonksiyonun örnekleme periyoduna bağlı olarak σ değeri bulunabilir. Bu tezde $T=1$ ve $\sigma=0.506$ değerleri kullanılmıştır.

Orjinal fonksiyonun örnek değerleri hücresel ağın girişlerine satır düzeninde verilmektedir.

$$V_{uij} = x(j) \quad (6.9)$$

(6.9) ifadesinde de görüleceği gibi aynı sütundaki hücrelerin giriş gerilimleri de aynıdır. Hücresel ağların giriş gerilimleri üzerindeki (3.2e) kısıtlamasına uymak için örnek değerleri devreye verilmeden önce 1'e normalize edilmelidir. Bir hücrenin çalışmasını genel olarak şöyle özetleyebiliriz: Her hücre etrafındaki komşuların kendi örnek noktasına yapacakları potansiyel etkinin toplamını hesaplar ve bu toplamı gerçek değerden çıkarır. Komşuların toplam etkisi orjinalden az ise kendi potansiyelini artırır ve çıkışı +1 değerine oturur, aksi takdirde kendi potansiyelini azaltır ve çıkışı -1'e yerleşir. Hücrelerin başlangıç koşulları her zaman

$$V_{xij}(0) = 0$$

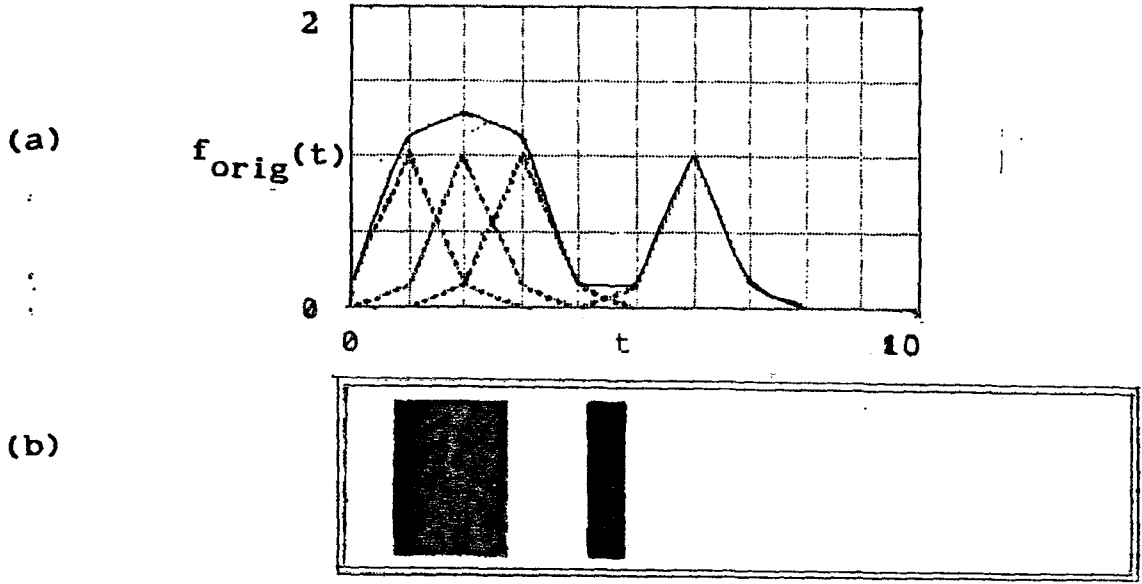
alınmalıdır. Negatif ayrıştırma kalıplarında polariteler tamamen ters olarak alınmalıdır. 5xN boyutlu bir hücresel ağ için pozitif değerli fonksiyonları ayrıştıran küme kalıbı şöyle verilebilir :

$$A = \begin{bmatrix} -0.1 & -1 & -0.1 \\ -0.1 & 2 & -0.1 \\ -0.1 & -1 & -0.1 \end{bmatrix} \quad B=b=5 \quad I=-2 \quad (6.10)$$

(6.10) küme kalıbıyla ayrıştırılmış bir fonksiyon Şekil 6.3'de yer almaktadır.

(6.10) küme kalıbı sadece pozitif değerli fonksiyonları ayrıştırmakta ve fonksiyonda negatif değerler varsa bu noktalar için 0 çıkışı oluşturmaktadır. Negatif işaretli fonksiyonları ayrıştırmak için küme kalıbında küçük bir değişiklik yapılarak aşağıdaki hale getirilir :

$$A = \begin{bmatrix} -0.1 & -1 & -0.1 \\ -0.1 & 2 & -0.1 \\ -0.1 & -1 & -0.1 \end{bmatrix} \quad B=b=5 \quad I=2 \quad (6.11)$$



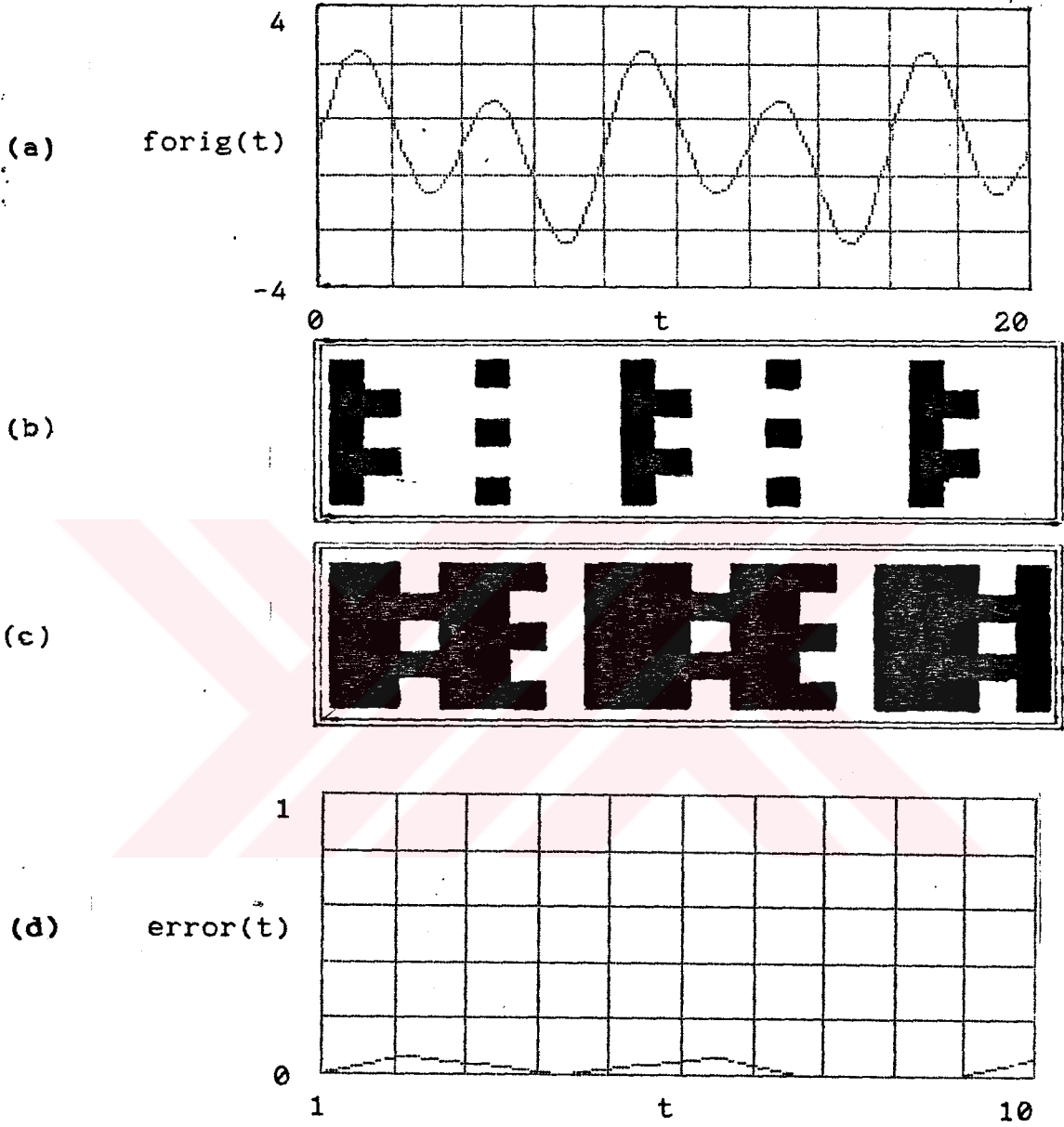
Şekil 6.3 HNA İşaret Ayırıştırma Örneği

a) 4 Gauss Fonksiyonunun Toplamı

b) HNA Çıkışı

(6.11) kalıbının kullanılmasıyla çıkışın polaritesi de değişir. -1 çıkışı veren hücrelere ait fonksiyonlar çözümde var, +1 olanlar ise çözümde yoktur.

İçerisinde hem pozitif hem de negatif değerler olan bipolar fonksiyonlar da (6.10) ve (6.11) küme kalıpları ile ayrıştırılabilir. Giriş değerleri pozitif ve negatif küme kalıplarıyla ayrı ayrı işlenir ve çıkışlar elde edilir. Bu çıkışlarda var olan Gauss fonksiyonlarına göre yaklaşık fonksiyonlar da gene ayrı ayrı hesaplanır. Negatif ayrıştırmadan sonra baz fonksiyonu olarak $-\epsilon_{\mu}(t)$ kullanılacağına dikkat edilmelidir. Negatif ve pozitif ayrıştırmalardan elde edilen bu fonksiyon değerleri toplanarak $f_{est}(t)$ fonksiyonumuzu oluştururlar. Şekil 6.4'de bir sinüs fonksiyonun bu şekilde ayrıştırılmasıyla elde edilen sonuç görülmektedir.



Şekil 6.4 HNA ile Bipolar Ayırıştırma

a) $f(t) = \sin(\omega t) + 2\sin(\omega t)$

b) HNA Pozitif Ayırıştırma Sonucu

c) HNA Negatif Ayırıştırma Sonucu

d) Mutlak Hata

Bipolar fonksiyonların ayrıştırılması düşeyde ötelenerek de yapılabilir. Fonksiyon ötelenerek pozitif değerli hale getirilir ve sadece pozitif ayrıştırma kalıbı kullanılarak ayrıştırılır. Doğal olarak öteleme işleminden sonra fonksiyonun tekrar normalize edilmesi gereklidir. Bipolar ayrıştırma ve öteleme yöntemlerinden hangisinin daha iyi sonuç vereceği ayrıştırılan fonksiyonun karakteristiğine bağlıdır. Bu iki yöntemin bir cosinüs fonksiyonu üzerinde karşılaştırılması Şekil 6.5'de yer almaktadır.

Ayrıştırma sonucunda elde edilen $f_{est}(t)$ ile gerçek fonksiyon arasındaki fark hatayı meydana getirir. Hata değeri kullanılan hücresel ağı satır sayısına bağlı olarak değişmektedir. Giriş gerilimlerini (3.2e) koşulunu sağlayacak şekilde normalize ettiğimize göre örneğin 5 satırlı bir ağda, tek bir Gauss fonksiyonunun genliği 0.2V'a karşı düşmektedir. Satır sayısını 10'a çıkartırsak tek bir Gauss fonksiyonunun genliği 0.1V'a iner ve daha hassas bir sonuç elde ederiz. Satır sayısı arttırıldığında küme kalıplarında küçük değişiklikler yapmak gerekir. 10 satırlı bir hücresel ağ için pozitif ve negatif küme kalıpları aşağıda verilmiştir:

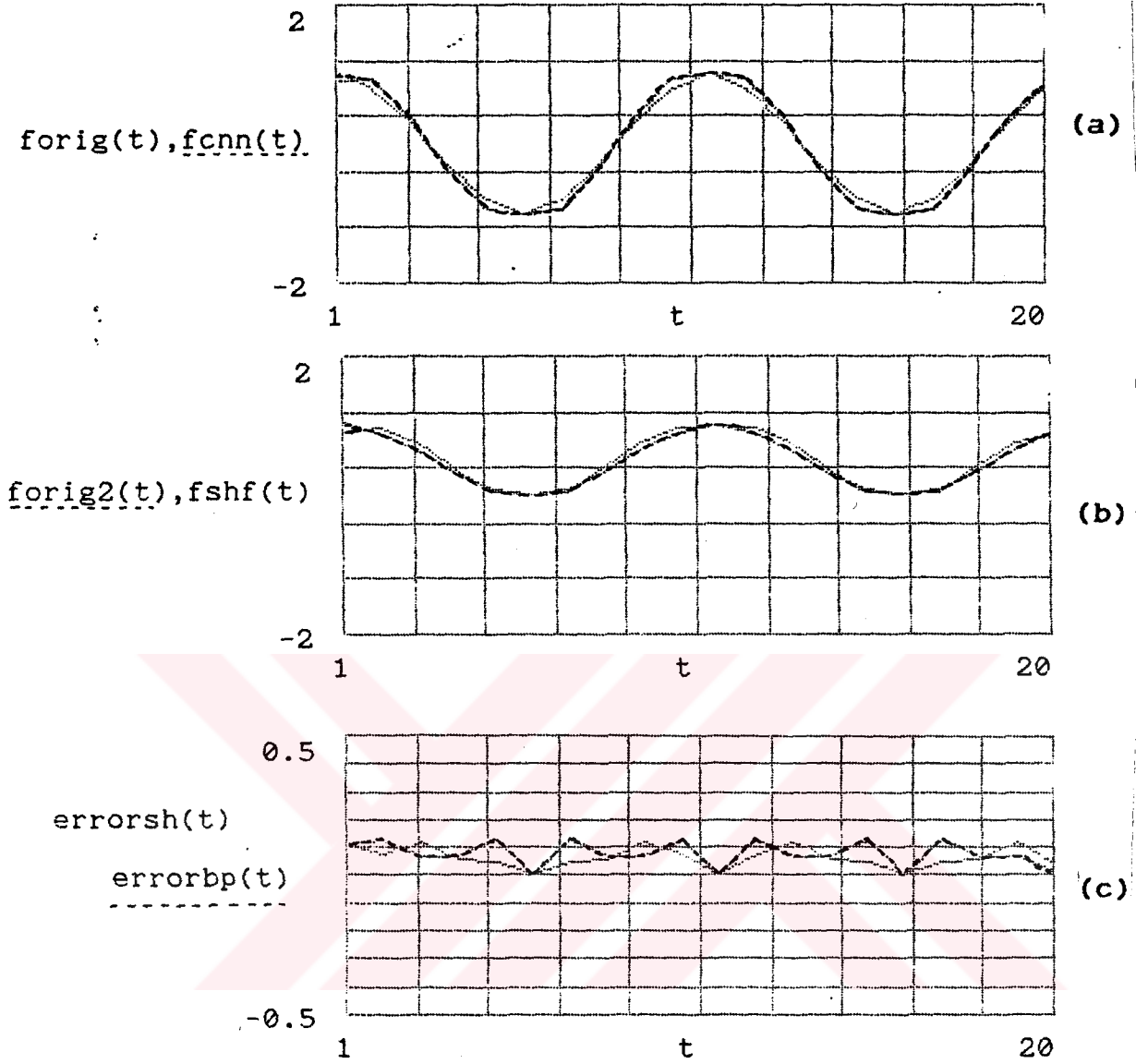
10 Satırlı Pozitif Ayrıştırma Kalıbı:

$$A = \begin{bmatrix} -0.1 & -1 & -0.1 \\ -0.1 & 2 & -0.1 \\ -0.1 & -1 & -0.1 \end{bmatrix} \quad B=b=10 \quad I=-5 \quad (6.12)$$

10 Satırlı Negatif Ayrıştırma Kalıbı:

$$A = \begin{bmatrix} -0.1 & -1 & -0.1 \\ -0.1 & 2 & -0.1 \\ -0.1 & -1 & -0.1 \end{bmatrix} \quad B=b=10 \quad I=5 \quad (6.13)$$

5 ve 10 satırlı ağların karşılaştırması Şekil 6.6 da yer almaktadır.



$$HATABP = \sum_{t=1}^{20} errorbp(t)^2 = 0.159$$

$$HATASH = \sum_{t=1}^{20} errorsh(t)^2 = 0.113$$

Şekil 6.5 Yatayda Öteleme ve Bipolar Ayırıştırma Farkı

a) $f(t) = \cos(wt)$ b) HNA Öteleme Cevabı

b) HNA Bipolar Ayırıştırma Cevabı

c) Mutlak Hataların Karşılaştırılması

forig(t)

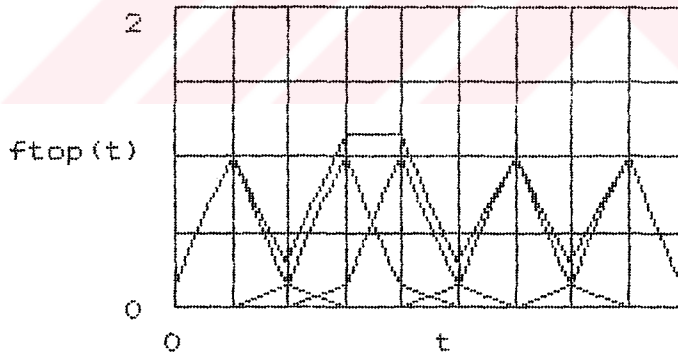
0.522
0.932
0.59
1
1
0.59
0.932
0.59
0.932
0.522
0.455
0.522
0.932
0.522
0.455
0.454
0.454
0.454
0.454
0.454

fcnn5(t)

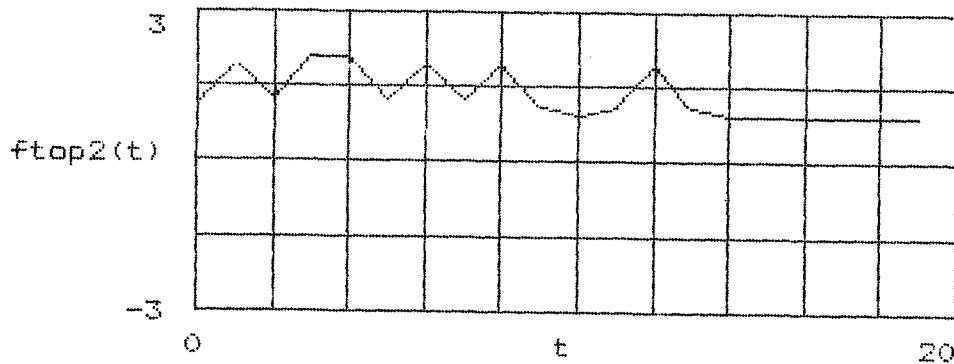
0.605
0.954
0.72
1
1
0.72
0.954
0.72
0.954
0.674
0.628
0.674
0.954
0.674
0.628
0.628
0.628
0.628
0.628
0.558

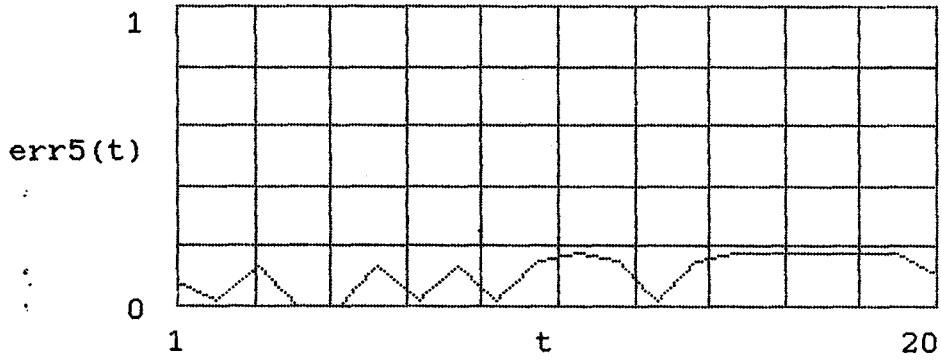
fcnn10(t)

0.605
0.954
0.72
1
1
0.72
0.954
0.72
0.954
0.651
0.465
0.651
0.954
0.651
0.442
0.419
0.419
0.419
0.419
0.372

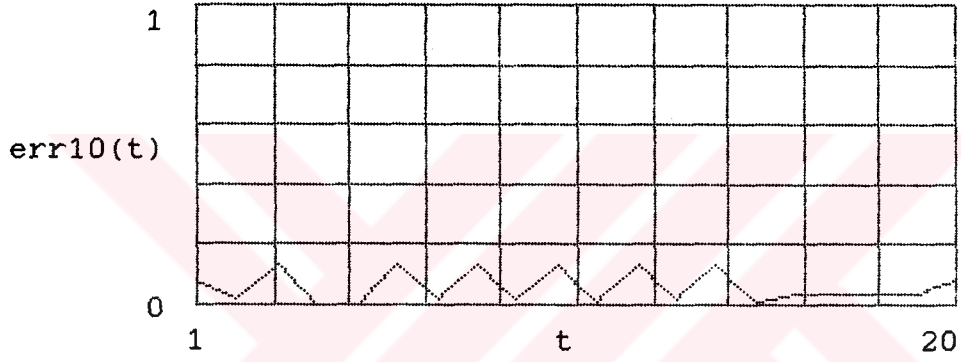


$ftop2(t) := ftop(t) + \sin(2 \cdot \pi \cdot 0.2)$





b)



c)

$$ER5 = \sum_{t=1}^{20} \text{err5}(t)^2 = 0.32$$

d)

$$ER10 = \sum_{t=1}^{20} \text{err10}(t)^2 = 0.121$$

Şekil 6.6 5 ve 10 Satırlı HNA Karşılaştırması

- a) Orjinal Fonksiyon ve HNA cevaplarının sayısal değerleri
- b) 5 Satırlı HNA Mutlak hatası
- c) 10 Satırlı HNA Mutlak Hatası
- d) Hatanın Karşılaştırılması

SONUÇ

Bu tezde Hücresel Nöral Ağ mimarisi ayrıntılı olarak incelenmiş ve uygulamalarından örnekler verilmiştir. Günümüze kadar literatüre geçmiş HNA uygulamaları çoğunlukla görüntü işleme ve tanıma konusuna yöneliktir. Özgün bir çalışma olarak bu tezde, hücresel ağ mimarisi işaret işleminin önemli problemlerinden biri olan İşaret Ayırıştırma problemine uygulanmıştır. N adet noktada örneklenmiş analog, bipolar işaretlerin, Gauss fonksiyonları baz alınarak bileşenlerine ayrılmasına dayana bu çalışmada, hücresel ağlar için 1-Komşuluk yapısında 4 adet yeni küme kalıbı bulunmuştur. Ayırıştırma işleminin çözünürlüğü kullanılan hücresel ağın satır sayısına göre değişmektedir. Bu tezde 5 satırlı ve 10 satırlı hücresel ağlar için ayırıştırma küme kalıpları verilmiştir. 5 satırlı ağların çözünürlüğü 0.2V, 10 satırlı ağların çözünürlüğü ise 0.1V olmaktadır. Devrenin doğruluğunu arttırmak için komşuluk sayısı ikiye çıkarılmak istenmiş ve aynı mantık kullanılarak aşağıdaki küme kalıbı elde edilmiştir.

$$A = \begin{bmatrix} -0.017 & -0.36 & -1 & -0.36 & -0.017 \\ -0.017 & -0.36 & -1 & -0.36 & -0.017 \\ -0.017 & -0.36 & 4 & -0.36 & -0.017 \\ -0.017 & -0.36 & -1 & -0.36 & -0.017 \\ -0.017 & -0.36 & -1 & -0.36 & -0.017 \end{bmatrix} \quad B=1 \quad I=-4 \quad (14)$$

Bu küme kalıbı karşılıklı olup kararlılık kriterlerini sağlamasına rağmen bilgisayar simülasyonlarında çıkışların $\bar{F}_1$ 'e yakınsamadığı ve devrenin de kararlı bir denge durumuna ulaşamadığı gözlenmiştir.

KAYNAKLAR

- [1] ROSKA T., "A dual way of computing: Learning from cerebral assymetry", Proc. Symp. Computer & Cognitive Sciences, Tampere, 1989.
- [2] KOHONEN, T., "Adaptive, associative and self organizing functions in Neural Computing", Applied Phys., Vol.26, No 23, pp:4910-4918, 1987.
- [3] McCULLOCH W.S, PITTS W.A, "A Logical Calculus of the Ideas Immanent in Nervous Activity", Bull. Math. Biophysics, Vol 5, pp:115-133, 1943.
- [4] McCULLOCH W.S, PITTS W.A, "How we know universals: The perception of auditory and visual forms", Bull. Math. Biophysics, Vol 9, pp:127-147, 1947.
- [5] ROSENBLATT, F., "The Perceptron: A probabilistic model for information storage and organization in the brain", Psychological Review, 65, pp:386-408, 1958.
- [6] MINSKY, M., PAPERT, S., Perceptrons, Cambridge, MA, MIT Press, 1969.
- [7] HOPFIELD, J., "Neural networks and physical systems with emergent collective computational abilities", Proc. Nat. Academy of Sci., 79, pp:2554-2558, 1982.
- [8] HOPFIELD, J., "Neurons with graded response have collective computational abilities like those of two state neurons", Proc. Nat. Academy of Sci., 81, pp:3088-3092, 1984.
- [9] HOPFIELD, J., Tank, D.W., "Computing with neural nets: A Model", Science, Vol 233, pp:625-632, 1986.
- [10] SPRECHER, W.A., "On the structure of continuous functions of several variables", Trans. Am. Math. Soc., Vol 115, pp:340-355, 1965.

- [11] HECHT-NIELSEN, R., "Counterpropagation Networks", Proc. 1st. Int. Conf. NN, Vol 2, pp:19-22, 1987.
- [12] WIDROW B., LEHR M., "30 Years of Adaptive Neural Networks: Perceptron, Madaline and Backpropagation", Proc. IEEE, Vol. 78, No 9, pp:1415-1442, 1990.
- [13] KOHONEN, T., "Self Organizing Map", Proc. IEEE, Vol 78, No 9, pp:1464-1480, 1990.
- [14] AMARI, S., "Mathematical foundations of Neurocomputing", Proc. IEEE, Vol 78, No 9, pp:1443-1463, 1990.
- [15] DIMOPULOS, N., "A study of the asymptotic behaviour of Neural Networks", IEEE T-CAS, Vol 36, No 5, pp:687-694, 1989.
- [16] MEAD, C., "Neuromorphic Electronic Systems", Proc. IEEE, Vol 78, No 9, pp:1629-1639, 1990.
- [17] CHUA L., YANG L., "Cellular Neural Networks: Theory", IEEE T-CAS, Vol 35, pp:1257-1272, 1988.
- [18] CHUA L., YANG L., "Cellular Neural Networks: Applications", IEEE T-CAS, Vol 35, pp:1273-1290, 1988.
- [19] HAHN, W., Theory and Application of Liapunov's Direct Method, Prentice Hall Int., Englewood Cliffs, N.J., 1963.
- [20] ROSKA, T., CHUA, L., "Cellular Neural Networks with nonlinear and delay type template elements" Proc. 1st Int CNNA Workshop, Budapest, pp:12-25, 1990.
- [21] ROSKA, T., CHUA, L., "Stability of a class of nonreciprocal cellular neural networks", Memorandum UCB/ERL M89/100, Univ. California, Berkley, 1989.

- [22] VANDENBERGHE, I., VANDEWALLE, J., "Finding multiple equilibrium points of cellular neural networks without enumeration", Proc. 1st Int CNNA Workshop, Budapest, pp:45-54, 1990.
- [23] GROSSBERG, S., "Competition, decision and consensus", Jr. Math. Anal. Appl., Vol 66, pp:470-493, 1978.
- [24] HIRSCH, M., "Convergence in Neural Nets", Proc. 1st. Int. Conf. NN, Vol 2, pp:115-125, 1987.
- [25] HIRSCH, M., "Systems of differential equations that are competitive or cooperative II: Convergence almost everywhere", SIAM Jr. Math. Anal., Vol 16, No 3, pp:423-439, 1985.
- [26] HECHT-NIELSEN, R., Neurocomputing, Addison-Wesley Co, Inc., 1990.
- [27] HALONEN, K., et al., "VLSI Implementation of a reconfigurable CNN containing local logic", Proc. IEEE First Int. Workshop of CNNA, pp:206-210, Budapest, 1990.
- [28] SPAANNENBURG, L., et al., "ASIC based development of CNN", Proc. IEEE First Int. Workshop of CNNA, pp:177-184, Budapest, 1990.
- [29] MATSUMOTO, T., CHUA, L., SUZUKI, H., "CNN Cloning Template: Connected Component Detector", IEEE T-CAS, Vol 37, No 5, pp:633-635, May 1990.
- [30] MATSUMOTO, T., CHUA, L., SUZUKI, H., "CNN Cloning Template: Shadow Detector", IEEE T-CAS, Vol 37, No 8, pp:1070-1073, August, 1990.
- [31] MATSUMOTO, T., CHUA, L., SUZUKI, H., "CNN Cloning Template: Hole Filler", IEEE T-CAS, Vol 37, No 5, pp:635-638, May 1990.
- [32] MATSUMOTO, T., CHUA, L., YOKOHAMA, H., "Image thinning with a cellular network", IEEE T-CAS, Vol 37, No 5, pp:638-640, May, 1990.

- [33] MATSUMOTO, et al., "Several Image Processing Examples by CNN", Proc. IEEE First Int. Workshop of CNNA, pp:101-111, Budapest, 1990.
- [34] ROSKA, T., BARTFAI, G., "A New Cellular Neural Network Template and its Application to Skeletonization", under preparation.
- [35] SEILER, G., "Small Object Counting with Cellular Neural Networks", Proc. IEEE First Int. Workshop of CNNA, pp:115-123, Budapest, 1990.
- [36] ROSKA, T., et al., "Detecting Simple Motion Using Cellular Neural Networks", Proc. IEEE First Int. Workshop of CNNA, pp:127-138, Budapest, 1990.
- [37] TANK, D., HOPFIELD, J., "Simple Neural Optimization Networks: An A/D Converter, Signal Decision Circuit and a Linear Programming Circuit", IEEE T-CAS, Vol 33, No 5, pp:533-541, May 1986.
- [38] UZALP, B., Neural Networks and a Hopfield Model Realization, Yüksek Lisans Tezi, Boğaziçi Üniversitesi, 1991.
- [39] ÖZMETELER, E., Yapay Nöral Ağlar, Yüksek Lisans Tezi, İTÜ, 1989.
- [40] Darpa Neural Network Study, AFCEA International Press, 1988.

ÖZGEÇMİŞ

Sevilay Özdemir 1967 yılında İstanbul da doğdu. Orta ve lise eğitimini Kadıköy Kız Lisesinde tamamladıktan sonra 1984 yılında bu liseden biricilikle mezun oldu. Aynı yıl İstanbul Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Kontrol ve Bilgisayar Müh. bölümüne giren Özdemir, 1988 yılında mezun olarak aynı bölümde lisans üstü eğitimine başladı. S.Özdemir master eğitimi boyunca Empa A.Ş Araştırma Bölümünde çalışmıştır.

W. C.

Yükseköğretim Kurulu
Dokümantasyon Merkezi