

DİNAMİK PARTİ BÜYÜKLÜĞÜ PROBLEMİ İÇİN YAPAY SINIR AĞI MODELLERİ

Yrd. Doç. Dr. Tuba YAKICI AYAN¹

Özet

Yapay sinir ağları (YSA) daha önce üretim planlama sürecindeki çeşitli problemleri çözmek için başarıyla kullanılmıştır. Bu makalede dinamik parti büyüklüğü problemi için biri reel diğeri 0-1 kodlamalı iki ayrı YSA geliştirilmektedir. Çalışmada geriye yayımlı öğrenme algoritması ve ileri beslemeli ağ yapısı kullanılmaktadır. Sonuçlar, her iki YSA modelinin dinamik parti büyüklüğü problemini çözmede son derece başarılı olduklarını ortaya koymaktadır.

Anahtar Kelimeler: *Yapay sinir ağı, geriye yayılım, ileri besleme, dinamik parti büyüklüğü problemi.*

NEURAL NETWORK MODELS FOR THE DYNAMIC LOT SIZING PROBLEM

Abstract

Artificial neural networks (ANN) have been used successfully to solve many problems in production planning process. In this paper, two neural network models - one of which is with reel coding and the other one is with binary coding - are developed to dynamic lot sizing. Back-propagation learning algorithm and feed-forward multi-layered architecture is used. Results show that both models are capable of solving the dynamic lot sizing problem with success.

Keywords: *Neural network models, back propagation, feed forward, dynamic lot sizing problem*

¹ KTÜ, İİBF, Ekonometri Bölümü, ayan@ktu.ed.tr

Giriş

Dinamik parti büyüklüğü problemi, bir planlama döneminin ardışık periyotlarında önceden bilinen değişken taleplere sahip bir tek ürün için, sipariş ya da üretim miktarlarının toplam maliyet en küçük olacak şekilde belirlenmesi problemidir. Optimum parti büyüklüğü, malzeme ihtiyaç planlama sistemlerinin en önemli elemanlarından biridir. Sabit talepler varsayımı altında ekonomik parti büyüklüğü için bir tek değer kolayca hesaplanabilmektedir² Fakat talepler planlama dönemi boyunca değişken olduğunda problem artık dinamik bir yapı kazanmakta ve her periyodun ayrı ayrı değerlendirilmesi gerekmektedir. Problemin çözümü için çeşitli sezgisel ve matematiksel yöntemler önerilmiştir. Bunlar arasında optimal çözümü garanti eden tek algoritma Wagner ve Whitin dinamik programlama eşaslı algoritmasıdır.³ Ancak bu algoritma öncelikle Silver ve Meal⁴ ve Peterson ve LaForge⁵, daha sonra ise başka araştırmacılar tarafından tarafından, uzun ve karmaşık hesaplamalar gerektirdiği yönünde eleştirilere konu olmuştur. Söz konusu eleştiriler sonucunda değişken taleplerle parti büyüklüğünün belirlenmesi probleminin çözümüne yönelik çeşitli sezgisel yöntemler geliştirilmiştir. Söz konusu sezgisel yöntemlerden en yaygın olarak kullanılanlar; Kısmi periyot dengeleme⁶, Silver – Meal⁷, TOPS⁸ ve MINS⁹ algoritmaları olarak özetlenebilir. Aggarwal ve diğerleri parti büyüklüğü probleminin çeşitli versiyonlarına dair kapsamlı bir literatür sunmaktadır¹⁰.

Parti büyüklüğü problemini çözmek için kullanılacak araçlardan birisi de yapay sinir ağları (YSA) dır. YSA insan beyninin öğrenme ve hatırlama özelliklerinden yararlanarak oluşturulmuş bir bilgi işleme sistemidir. Bir YSA ayarlanabilir ağırlıklarla birbirine bağlı birçok basit işlem biriminden oluşmaktadır. Geleneksel model esaslı metotların aksine YSA lar değişkenler arasındaki ilişkiler bilinmese veya anlaşılmalrı zor olsa bile bu ilişkileri veriler arasından yakalayabilmekte ve örneklerden öğrenebilmektedirler. YSA lar genelleme yapma, bir veri kümesindeki yapıyı kavrama, sınıflama ve değişen şartlara uyum sağlama yeteneklerine sahip olan alternatif hesaplama araçlarıdır Ayrıca hızlı çıkarım yapma ve yüksek çalışma etkinliği gibi avantajlara sahip

² Whitin, T. M., **The Theory of Inventory Management**, 2nd ed., Princeton University Press, Princeton, NJ., 1957

³ Wagner, H.M. and Whitin, T.M., “Dynamic Version of the Economic Lot Size Model”, **Management Science**, 2004, 50(12), s. 1770-1774

⁴ Silver, E.A. and Meal, H.C., “A Heuristic for Selceting Lot Size Quantities for the Case of Deterministic Time-Varying Deman Rate and Discrete Opportunities for Replenishment”, **Production Inventory Management**, 1973, 14 (2), s. 64-74.

⁵ Patterson, J.W., LaForge, R.L., “The Incremental Part Period Algorithm: an Alternative to EOQ”, **Journal of Purchase and Material Management**, 1985, 21 (2), s. 28-33.

⁶ DeMatteis, J.J., “An Economic Lot Sizing Technique I: The Part Period Algorithm”, **IBM System Journal**, 7(1), 1968, s. 39-48.

⁷ Silver, E.A. and Meal, H.C., a.g.m.

⁸ Coleman, B.J., McKnew, M.A., “A Technique for Order Placement and Sizing”, **Journal of Purchase and Materials Management**, 1990, 26(1),s. 32-40.

⁹ Zhiwei, Z, Heady, R.B., Lee, J., “A Simple Procedure for Solving Single Level Lot Sizing Problems”, **Computers & Industrial Engineering**, 1994, 26 (1), s. 125-131.

¹⁰ Aggerwal, A. and Park, J.K., “Improved Algorithms for Economic Lot Size Problems”.**Operations Research**, 1993, 41 (3), s. 549-571

olmalarından dolayı YSA lar çeşitli üretim planlama problemlerinde başarılı uygulamalara sahiptirler. Bunlardan bazıları, Ntuen¹¹, Laarhoven¹², Yih ve diğerleri¹³, Brace ve diğerleri¹⁴ ve Kim ve diğerleri¹⁵ tarafından yapılan YSA uygulamalarıdır.

Üretim planlama amacı ile geliştirilen ağlara parti büyüklüğü probleminin de dahil edilmesi üretim bilgi sisteminin bütünleştirilmesine imkan vererek çok kullanışlı bir araç oluşturabilir Ancak literatüre bakıldığında parti büyüklüğü problemi için çok az sayıda YSA çalışması ile karşılaşmaktadır. Zwietering ve diğerleri uygun şekilde oluşturulmuş ve iyi eğitilmiş çok katmanlı bir YSA nın Wagner - Whitin problemini çözmede geleneksel algoritmalarından daha başarılı olacağını ortaya koymuşlardır¹⁶. Ezziane ve diğerleri bir envanter yönetimi sisteminde hammadde siparişi verip vermeme kararı için YSA yaklaşımını kullanmışlardır.¹⁷ Stehouwer ve diğerleri parti büyüklüğü problemine fazla mesaide üretim maliyetlerini de dahil ederek bir YSA çalışması yapmışlardır.¹⁸ Gaafar ve Choueiki parti büyüklüğü problemi için oluşturulan YSA modelinin performansını problemi çözmek için yaygın olarak kullanılan bazı sezgisel yöntemlerle karşılaştırmışlardır.¹⁹ Son olarak yeni bir performans karşılaştırması, Gaafar ve Choueiki'ninkinden çok daha küçük bir örnek hacmi ile (3000 e karşı 70) Radzi ve diğerleri tarafından yapılmıştır.²⁰

Bu çalışmada dinamik parti büyüklüğü problemini çözmek için yapay sinir ağları kullanılmaktadır. Daha önce bu problem için Radzi ve diğerleri²¹ ve Gaafar ve Choueiki²² tarafından yapılmış olan YSA çalışmalarında. çözüm sonuçları, herhangi bir periyotta üretim (veya sipariş) olup olmadığını göstermek üzere 0-1 değişken olarak kodlanmıştır. Ancak 0-1 kodlama üretim (sipariş) miktarlarını belirlemek üzere sonuçların yeniden yorumlanmasını gerektirmektedir. Bu çalışmada 0-1 kodlamanın yanı sıra çıktılardan

¹¹ Ntuen, CA. "A Neural Network Model for a Holistic Inventory System", **International Industrial Engineering Conference**, s. 435-444, 1991

¹² Laarhoven, P.J., Aarts, E.H, Lenstra, J.K., "Job Shop Scheduling by Simulated Annealing", **Operations Research**, 1991, 40, s. 113-125

¹³ Yih, Y., Liang, T., Moskowit, H., "Robot Scheduling in a Circuit Board Production Line: a Hybrid OR/ANN Approach", **IIE Transactions**, 1993, 25, s. 26-33

¹⁴ Brace, M.C, Bui-Nguyen, V., Schmidt, J., "Another Look at Forecast Accuracy of Neural Networks", **In: Proceedings of the Second Joint Forum on Applications of Neural Networks to Power Systems**, Yokohama, Japan, s. 19-22, 1993.

¹⁵ Kim, S., Lee, Y., Agnihotri, D., "A Hybrid Approach to Sequencing Jobs Using Heuristic Rules and Neural Networks", **Production Planning and Control**, 1995, 6(5), s. 445-454.

¹⁶ Zwietering, P.J., Kraaij, M.J.A.L., Aarts, E.H.L., Wessels, J., "Neural Networks and Production Planning", **In Proceedings of the Fourth International Conference on Neural Networks and their Applications**, s. 529-542, 1991.

¹⁷ Ezziane, Z.H., Mazouz, A.K., Han, C., "Neural Network Approach for Inventory Control", **Intelligent Robotics and Computer Vision XI. SPIE**, 1992, s. 501-9.

¹⁸ Stehouwer, H.P., Aarts, E.H.L., Wessels, J. "Multi Layered Perceptrons for on-line Lot Sizing", Technical Report, The Netherlands: Eindhoven University of Technology, ss. 279-287, 1995.

¹⁹ Gaafar, L.K. and Choueiki, M.H., "A Neural Network Model for Solving The Lot Sizing Problem", **Omega**, 28, 2000, s. 175-184.

²⁰ Radzi, N. H. M., Haron, H., Irdawati, T., Johari, T., "Lot Sizing Using Neural Network Approach", **Regional Conference on Mathematics, Statistics and Applications**, Malaysia, ss. 1-8, 2006.

²¹ Radzi, N. H. M., Haron, H., Irdawati, T., Johari, T., a.g.m.

²² Gaafar, L.K. and Choueiki, M.H., a.g.m.

doğrudan üretim (sipariş) miktarlarını gösterdiği reel kodlamalı ikinci bir ağ kurulmaktadır. Çalışmanın bir amacı dinamik parti büyüklüğü problemini çözmede YSA'nın kullanımını ayrıntılı olarak açıklamak iken ikinci amaç ise 0-1 kodlamalı ve reel kodlamalı ağların performanslarını karşılaştırmaktır.

Çalışmanın birinci bölümünde birinci bölümünde dinamik parti büyüklüğü problemi açıklanmaktadır. İkinci bölümde probleme dair bir YSA geliştirilmektedir. Dördüncü bölüm uygulama ve bulguların değerlendirilmesine ayrılmıştır. Son bölüm çalışma sonuçlarını ve önerileri içermektedir.

1. Dinamik Parti Büyüklüğü Problemi

Bu çalışmada ele alınan, değişken taleplerle optimal parti büyüklüğü belirleme probleminde, belli bir periyoda ilişkin talep bu periyotta veya bir elde bulundurma maliyetine katlanarak daha önceki bir periyotta üretilebilir (veya sipariş verilebilir). Üretim veya siparişin özel bir periyodun hangi noktasında gerçekleştirildiği dikkate alınmamaktadır. Talep miktarları bölünemez yapıdadır. Diğer bir deyişle herhangi bir periyodun talebi yalnız bir periyotta üretilebilir (sipariş verilebilir). Ayrıca üretilen ürünün satış fiyatının veya sipariş edilen malzemenin satın alma fiyatının planlama dönemi boyunca sabit kaldığı varsayılmaktadır. Dolayısı ile toplam maliyet, parti başına üretim hazırlık masrafları (veya sipariş masrafları) ile birim stok bulundurma maliyetinin toplamından oluşmaktadır. T adet periyot için problem her bir periyotta üretip üretmeme (sipariş verip vermeme) şeklinde 2^{T-1} adet kombinasyon değerlendirmeyi gerektirmektedir. Dinamik parti büyüklüğü probleminin bir dinamik programlama formülasyonu aşağıdaki gibidir.²³

d_t = t periyodundaki talep miktarı

i_t = t periyodunda bir birim ürünü elde bulundurma maliyeti

s_t = t periyodunda parti başına üretim hazırlık (sipariş) maliyeti

x_t = t periyodundaki üretim (sipariş) miktarı

t periyodunda eldeki stok miktarı; $I_t = I_0 + \sum_{j=1}^{t-1} x_j - \sum_{j=1}^{t-1} d_j \geq 0$ (1.1)

Bu durumda t periyodu için en düşük maliyetli kararı gösteren amaç fonksiyonu ;

$$f_t(I_t) = \min_{\substack{x_t \geq 0 \\ I_t + x_t \geq d_t}} [i_{t-1}I_t + \delta(x_t)s_t + f_{t+1}(I_t + x_t - d_t)] \quad (1.2)$$

Burada $\delta(x_t)$, t periyodunda üretim yapılıp yapılmadığını (sipariş verilip verilmediğini) gösteren bir 0-1 değişkendir. Ayrıca son periyot için amaç fonksiyonu;

$$f_T(I_T) = \min_{\substack{x_T \geq 0 \\ I_T + x_T = d_T}} [i_{T-1}I_T + \delta(x_T)s_T] \quad (1.3)$$

²³ Wagner, H.M. and Whitin, T.M., a.g.m.

2. Yapay Sinir Ağları

Yapay sinir ağları, belli girdiler ve bu girdilere karşılık gelen çıktılar arasındaki ilişkileri öğrenmek için insan beynini taklit eden bilgisayar sistemleridir. Temel olarak tek katmanlı ve çok katmanlı olarak iki gruba ayrılmaktadırlar. Tek katmanlı ağlarda sadece girdi ve çıktı katmanları bulunmaktadır. Çok katmanlı ağlar ise girdi ve çıktı katmanlarına ilaveten en az bir ara (gizli) katman içermektedir. Her katman hücrelerden oluşmakta ve farklı katmanlardaki hücreler birbirlerine özel ağırlık değerleri ile bağlanmaktadır. Bu çalışmada ele alınan çok katmanlı ağların kurulması ve çalışması belirli aşamalardan oluşmaktadır. Söz konusu aşamalar: a) incelenecek olaya ait örneklerin toplanması, b) ağı geometrik yapısının belirlenmesi, c) öğrenme katsayısı, toplama ve aktivasyon fonksiyonları ve momentum katsayısı gibi öğrenme parametrelerinin belirlenmesi d) başlangıç ağırlık değerlerinin atanması, e) örneklerin ağa gösterilmesi f) ağı çıktılar üretmesi ve g) ağırlık değerlerinin adım adım iyileştirilmesi olarak özetlenebilir. Bu bölümde dinamik parti büyüklüğü problemi için kullanılacak bir yapay sinir ağı oluşturulması ve işleyişi açıklanmaktadır.

2.1. Eğitim ve Test Verisinin Oluşturulması

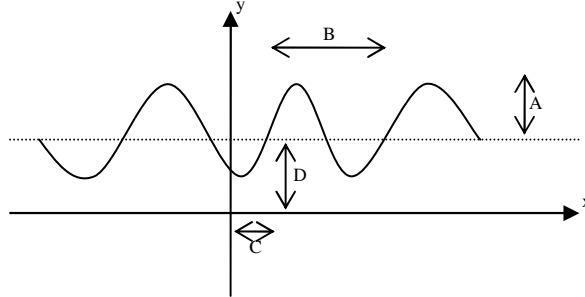
Oluşturulan yapay sinir ağında eğitim ve test amacı ile kullanılmak üzere sabit veya değişken talep yapılarına sahip olan 1500 örnek problem oluşturulmuştur. Örneklerin oluşturulmasında Excel programı kullanılmıştır. Problem girdileri parti başına üretim hazırlık maliyeti (veya sipariş maliyeti), birim stok maliyeti ve her bir periyot için sabit veya değişken taleplerden oluşmaktadır. Çıktılar ise planlama dönemindeki toplam maliyeti en küçük yapacak şekilde, ne zaman ve ne kadar sipariş verileceğini gösteren değişkenlerden ibarettir. Her bir problem için düzgün dağılımdan rasgele olarak 500 TL ile 4000 TL arasında sipariş maliyetleri (s_i) ve 1 TL ile 8 TL arasında birim stok maliyetleri (i_i) oluşturulmuştur. Planlama hattının her periyodunda değişken talep yapıları oluşturabilmek için şekil 1 de gösterilen ve genel formu eşitlik (2.1) de verilen sinüzoidal fonksiyon kullanılmıştır.

$$d_t = A * \sin\left(\frac{2\pi}{T}(t - C)\right) + D \quad (2.1)$$

Yukarıdaki eşitlikte D talebin beklenen değerini, A, beklenen talepten en büyük sapma miktarını ve T planlama hattının bölüdüğü periyot sayısını göstermektedir. C ise talep dalgalanmalarının yönünü belirlemektedir. Bu çalışmada periyot sayısı yer kısıtlamasından dolayı 6 olarak alınmıştır. 1500 adet farklı örnek oluşturmak için sinüzoidal fonksiyon parametrelerinin farklı kombinasyonları kullanılmıştır. Farklı kombinasyonlar elde edilirken D, 1000 ortalamalı üstel dağılımdan rasgele oluşturulmuştur. A parametresi için talebin sabit bir oranını kullanan benzer çalışmalardakinin aksine burada (0-1] aralığında düzgün dağılımlı bir rasgele sayı ile belirlenen rastlantısal bir oran kullanılmıştır. C ise [0-6] aralığında düzgün dağılımdan rasgele belirlenmiştir. Problemler oluşturulduktan sonra yapay sinir ağına çıktıları elde etmek üzere, Wagner-Within dinamik programlama esaslı algoritması kullanılarak Excel programı ile optimal olarak çözülmüşlerdir. Çözüm sonuçları iki farklı çıktı üretmiştir. Söz konusu çıktılarından biri her bir periyotta verilecek sipariş miktarlarının gerçek gösterimi iken diğeri ise her hangi bir periyotta sipariş verilip verilmeyeceğini ifade eden 0-1 kodlamadır. Oluşturulacak yapay sinir ağlarında yalnızca [0-1] aralığında değerler üretebilen sigmoid aktivasyon fonksiyonları kullanılacağı için

çıktıların da buna uygun olarak aynı aralıkta ölçeklendirilmeleri gerekmektedir. Ayrıca aşırı küçük veya aşırı büyük değerlerin ağı yanlış yönlendirme olasılığını yok etmek için girdi setleri de ölçeklendirilmektedir. Ölçeklendirme eşitlik (2.2) ye göre yapılmıştır.

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (2.2)$$



Şekil 1 Sinüzoidal fonksiyonun grafiği

2.2. Yapay Sinir Ağının Oluşturulması

Çok katmanlı yapay sinir ağları bir girdi katmanı, bir çıktı katmanı ve girdi ile çıktı arasındaki karşılıklı etkileşimleri temsil eden bir veya daha fazla ara (gizli) katmandan ibarettirler. Her katman çok sayıda yapay sinir hücresi içermekte ve hücreler arasında belirli ağırlık değerlerine sahip olan bağlantılar yer almaktadır. Yapay sinir ağının eğitim süreci esasen bağlantı ağırlıklarının problem amacını gerçekleştirecek şekilde belirlenmesi olarak özetlenebilir. Bir ağın iyi bir performans sergileyebilmesi için öncelikle geometrisinin en uygun şekilde belirlenmesi gerekmektedir. Girdi ve çıktı katmanlarında yer alacak sinir hücrelerinin sayısı probleme özgüdür. Dolayısı ile ara katmanların sayısı ve bu katmanlarda yer alacak hücre sayıları araştırmacı tarafından belirlenmektedir.

Ara katman sayısının belirlenmesi için standart bir yöntem bulunmamakta ve araştırmacının deneyimleri veya deneme yanılma yolu kullanılmaktadır. Literatürde en çok karşılaşılan ağlar tek veya iki ara katmanlı olanlardır. Ara katmandaki sinir hücresi sayısı da genellikle deneyim veya deneme yanılma yolu ile belirlenmekle birlikte literatürde bu konuda geliştirilmiş bazı yöntemler mevcuttur. Bunlardan başlıcaları, Hect-Nilsen, R.²⁴, Jadid and Fairbrain²⁵, Lachermacher and Fuller²⁶, Masters²⁷ ve Upadhaya and Eryureka²⁸

²⁴Hecht-Nielsen, R., **Neurocomputing. Reading**, Massachusetts, Addison-Wesley, 1990.

²⁵Jadid, M.N. and Fairbairn, D.R., "Predicting Moment Curvature Parameters from Experimental Data", **Engineering Application Artificial Intelligence**, 1996, 9 (3), s. 303-319.

²⁶Lachermacher, G. and Fuller, J.D., "Backpropagation in Time Series Forecasting", **Journal of Forecasting**, 1995, 14, s. 381-393

²⁷Masters, T., **Practical Neural Networks Recipes in C++**, Academic Press, San Diego, 1993.

nın yöntemleridir. Bu çalışmada tercih edilen Masters'ın yönteminde gizli hücre sayısı, girdi ile çıktı çarpımının karekökü olarak belirlenmektedir.

Yapay sinir ağları temel olarak ileri beslemeli ve geri beslemeli olarak ikiye ayrılmaktadır. İleri beslemeli ağlarda bir hücrenin çıktısı bir sonraki tabakadaki hücrelere girdi olarak sunulmaktadır. Geri beslemeli ağlarda ise bir hücrenin çıktısı tekrar kendisine girdi olarak geri dönebilmektedir. Bir tabakaya gelen net girdi, toplama fonksiyonu yardımı ile hesaplanmaktadır. Net girdi bu tabakada aktivasyon fonksiyonu ile dönüşüme tabi tutularak bir sonraki tabakaya gönderilmek üzere çıktı oluşturmaktadır. Çarpım, maksimum, minimum ve çoğunluk gibi farklı toplama fonksiyonları mevcut olmakla birlikte bu çalışmada girdilerin ağırlıklı toplamı tercih edilmektedir. Net girdiyi dönüştürmek üzere kullanılacak çeşitli aktivasyon fonksiyonları kullanılabilir. Ancak bu çalışmada kullanılan eğitim algoritması aktivasyon fonksiyonunun türevi alınabilir bir fonksiyon olmasını gerektirmektedir. Bu amaçla aşağıdaki sigmoid fonksiyonu tercih edilmiştir.

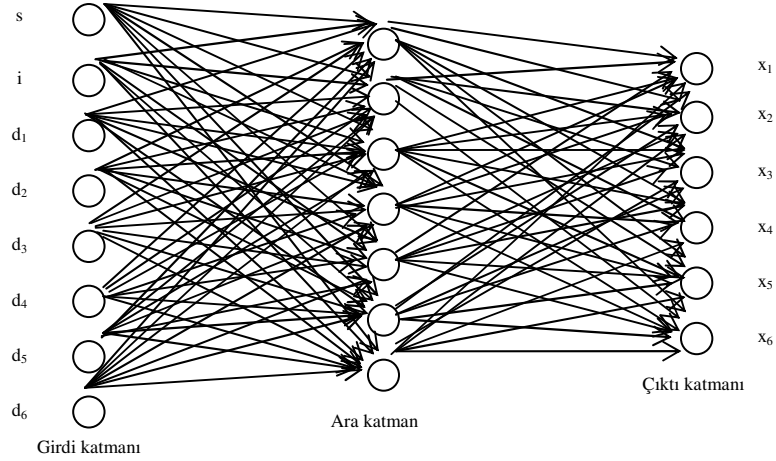
$$F(net) = \frac{1}{1 + e^{-net}} \quad (2.3)$$

Şekil 2 de gösterilen ileri beslemeli çok katmanlı yapay sinir ağının fonksiyonel formu aşağıdaki gibi ifade edilebilmektedir.

$$y_k = f_k \left(\alpha_k + \sum_{i=1}^7 w_i f_i \left(\alpha_j + \sum_{j=1}^8 \sum_{i=1}^8 w_{ij} x_j \right) \right) \quad k=1,2,\dots,6 \quad (2.4)$$

Burada y_k ağ tarafından üretilen son çıktıyı, w_i ara katman ile çıktı katmanı arasındaki ağırlıkları, w_{ij} girdi katmanı ile ara katman arasındaki ağırlıkları, x_j girdi değerlerini, f_k çıktı katmanındaki ve f_j ise ara katmandaki aktivasyon fonksiyonunu göstermektedir. α_k ve α_j eğitim hızını artırmak amacı ile tercihe bağlı olarak her bir hücreye eklenebilen eğilim parametreleridir. Girdi hücrelerinin eğilimleri (α_j) 1 olmak zorunda iken diğer eğilimler herhangi bir değer alabilirler.

²⁸ Upadhaya, B. and Eryureka, E., "Application of - Neural Networks for Sensory Validation and Plant Monitoring", **Neural Technology**, 1992, 97, s. 170-176.



Şekil2 İleri beslemeli çok katmanlı YSA

2.3. Eğitim Süreci

Çok katmanlı yapay sinir ağlarının işleyiş mekanizması, ağın incelenmekte olan olayı öğrenmek üzere eğitilmesi ve daha sonra öğrendiklerini benzer olaylarla karşılaştığında kullanması olarak açıklanabilir. Burada ağ kendisine gösterilen girdi ve çıktılar arasındaki ilişkiyi öğrenmektedir. Ağın görevi problem amacına en uygun bağlantı ağırlıklarını belirlemek ve bu ağırlıkları kullanarak daha önce görmemiş olduğu problemlere dair çıktıları üretmektir. Ağırlıklar eğitim aşamasının başlangıcında rasgele olarak belirlenmekte ve ağa örnekler sunuldukça aşamalı olarak iyileştirilmektedir. Çok katmanlı ağların öğrenme kuralı hata kareleri toplamının en küçüklenmesine dayalı delta kuralının genelleştirilmiş halidir. Bu kurala geriye yayım modeli denmekte ve iki aşamadan oluşmaktadır; 1) İleriye doğru ağ çıktıları hesaplama aşaması ve 2) Gerçek çıktı değerinden sapmaları diğer bir deyişle hataları geriye yayarak ağırlıkları değiştirme aşaması. Ağ ileriye doğru girdilerin ağırlıklı toplamlarını aktivasyon fonksiyonundan geçirerek hareket ederken geriye doğru hareketlerinde ise aktivasyon fonksiyonlarının türevlerini kullanmaktadır. Bu nedenle geriye yayım modelinde kullanılacak aktivasyon fonksiyonu herhangi bir türevlenebilir fonksiyon olmalıdır.

2.4. Çıktıların Hesaplanması

Bu aşama girdi katmanında bir örneğin ağa sunulması ile başlamaktadır. Girdiler rasgele seçilmiş ağırlık değerleri ile çarpılmakta ve bu çarpımların toplamı girdi katmanındaki hücrelerin net çıktıları dolayısı ile ara katmandaki hücrelerin net girdilerini oluşturmaktadır.

$$Net_j = \sum_{i=1}^8 x_i * w_{ij} \quad j = 1, 2, \dots, 7 \quad (2.5)$$

Ara katman hücrelerine gelen net girdiler eşitlik (2.3)deki aktivasyon fonksiyonundan geçtikten sonra gene ağırlıklı toplamları alınarak çıktı katmanına gönderilmektedir. Çıktı katmanı aldığı net girdileri son kez aktive ederek problem çıktılarını hesaplamaktadır. Bu aşamada üretilen çıktının hata miktarı hesaplanmakta ve yeni örnek ağa gösterilmeden önce ağırlıklar tekrar ayarlanmaktadır. Bütün örnekler ağa gösterilmiş olduğunda bir çevrim (epoch) tamamlanmış olmakta ve hata kareleri ortalaması hesaplanmaktadır. Eğitim süreci pek çok çevrimden oluşmaktadır.

Hata kareleri ortalamasının her çevrimde azalması ağıın öğrendiğinin bir göstergesi olmakla birlikte bu yeterli değildir. Eğitim sürecinde karşılaşılabilecek önemli bir sorun ağıın öğrenmeyi ezberlemesidir. Uygulamada ezberlemeyi kontrol ederek engelleme amacı ile veri setinin bir kısmı ayrılmakta ve her devirin sonunda her iki veri seti için hata kareleri ortalaması hesaplanmaktadır. Kontrol setinin hatasında değişme olmadığı halde eğitim setinin hatasının azalmaya devam ettiği nokta ağıın ezberlemeye başladığı noktadır. Bu noktada eğitime son verilmelidir.

2.5. Ağırlıkların Değiştirilmesi

Geri yayılım algoritmasında ara katman ile çıktı katmanı arasındaki ağırlık değerleri aşağıdaki formüle göre değiştirilmektedir.

$$w_j(t) = w_j(t-1) + \lambda * C_k * (1 - C_k) * \epsilon_k + \alpha * \Delta w_j(t-1) \quad (2.6)$$

Burada $w_j(t)$ t zamanında, ara katmanla çıktı katmanı arasındaki bir ağırlık değerini, C_k çıktı katmanındaki k hücresinin çıktısını ve ϵ_k gerçek çıktı değeri ile ağ çıktısı arasındaki sapmayı göstermektedir. λ , [0-1] aralığında belirlenen ve katsayıların değişim oranını ifade eden bir parametredir. α ise gene [0-1] aralığında olup ağıın öğrenme sırasında yerel bir optimuma takılıp kalmasını önlemek için kullanılmakta ve momentum katsayısı olarak adlandırılmaktadır. Benzer şekilde girdi katmanı ile ara katman arasındaki ağırlıkların yeni değerleri,

$$w_{ij}(t) = w_{ij}(t-1) + \lambda * C_j (1 - C_j) * C_i \sum C_k (1 - C_k) \epsilon_k w_j(t) + \alpha * \Delta w_{ij}(t-1) \quad (2.7)$$

şeklinde belirlenmektedir. Burada C_i ara katmandaki, C_j ise girdi katmanındaki bir hücrenin çıktısını göstermektedir.

2.6. Eğitimi Durdurma ve Ağıın Performansını Değerlendirme

Bir yapay sinir ağı probleme optimal çözüm üretecek ağırlıkları bulduktan sonra eğitime devam edilirse tekrar performansı düşük veya öğrenemeyen ağlara dönüşebilir. Bu nedenle eğitime nerede son verileceği son derece önemlidir. Uygulamada ezberlemenin başladığı nokta ile birlikte kullanılabilir iki ilave durdurma kriteri mevcuttur: 1) hata kareleri ortalamasının kabul edilebilir bir değer altına düşmesi, 2) belli bir çevrim sayısının tamamlanması. Kabul edilebilir hata miktarı probleme göre değişmekte ve araştırmacı tarafından belirlenmektedir. Eğitimin durdurulacağı çevrim sayısı, birkaç deneme yapıp hata grafikleri incelenerek belirlenebilir.

Bir ağ eğitildikten sonra artık kullanıma hazırdır ve ağıın performans göstergesi, daha önce hiç görmemiş olduğu örnekler karşısındaki davranışı ile belirlenebilir. Uygulamada hata miktarlarına dayanan bir çok performans ölçüsü mevcuttur. Ancak bu çalışmada ağıın doğru çıktı ürettiği test örneği yüzdesi kullanılmaktadır.

3. Uygulama

Çıktı katmanında biri gerçek üretim (sipariş) miktarlarını gösteren (reel) diğeri ise üretimin (siparişin) gerçekleşip gerçekleşmediğini gösteren iki ayrı YSA'nın eğitilmesi için PELTAIRON Synapse 1.3 paket programı kullanılmıştır. YSA larda kullanılan verilerin ilk 20 adedi Tablo 1 de sunulmaktadır. Tablodaki x_i sütunları problemlerin optimal çözüm sonuçlarını göstermektedir, 0-1 kodlama durumunda dolu olan x_i sütunları 1 boş olanlar ise 0 değerini almaktadır.

Tablo 1: YSA'yı Eğitmek ve Test Etmek İçin Kullanılan 1500 Örneğin İlk 20'si

s	i	d_1	d_2	d_3	d_4	d_5	d_6	x_1	x_2	x_3	x_4	x_5	x_6
573	6	1253	704	1197	807	1057	971	1253	704	1197	807	1057	971
3267	2	318	54	291	104	224	183	1174	0	0	0	0	0
2114	7	1087	5595	2029	4314	3527	2740	1087	5595	2029	4314	3527	2740
1216	2	3330	3724	3412	3612	3543	3475	3330	3724	3412	3612	3543	3475
2089	1	291	642	94	787	15	792	1027	0	0	1594	0	0
1356	4	112	182	158	134	203	95	294	158	134	0	298	0
1603	8	511	436	361	580	238	670	511	436	361	580	238	670
2536	8	99	1156	320	856	671	487	99	1156	320	856	671	487
2196	6	145	303	249	194	353	106	448	249	194	0	459	0
1643	3	268	268	267	269	266	270	803	0	269	266	270	0
3002	5	704	851	621	912	588	915	704	851	621	1500	0	915
3791	8	1923	1435	2853	641	3437	323	1923	1435	2853	641	3760	0
1911	2	392	824	150	1001	54	1008	1216	150	0	1055	0	1008
568	8	947	752	927	788	877	847	947	752	927	788	877	847
3784	6	307	238	169	369	57	451	714	0	0	426	0	451
1011	8	462	1158	608	961	839	718	462	1158	608	961	839	718
3056	6	463	428	393	495	336	537	891	393	0	831	0	537
824	2	2071	3059	2719	2379	3367	1825	2071	3059	2719	2379	3367	1825
3620	6	184	159	134	206	94	236	477	0	206	330	0	0
2508	1	473	131	350	275	199	419	1428	0	0	419	0	0

Bir kısmı test amacı ile ayrılan veri, eğitimin geçerliliğini doğrulamak amacı ile kullanılmak üzere tekrar iki parçaya bölünmüştür. Her bir YSA için eğitim parametreleri 10 ayrı deneme koşumu ile belirlenmiştir. Durdurma kriteri olarak hata kareleri toplamının karekökü kullanılmış ve kabul edilebilir hata düzeyi 0.01 olarak alınmıştır. Eğitim ve test sonuçları Tablo 2 de özetlenmektedir.

Tablo 2: YSA Eğitim ve Test Sonuçları

	Reel Kodlama	0-1 Kodlama
Eğitim verisi sayısı	1020	1020
Geçerlilik verisi sayısı	255	255
Test verisi sayısı	225	225
Girdi hücresi sayısı	8	8
Çıktı hücresi sayısı	6	5
Ara katman sayısı	1	1
Ara hücre sayısı	7	6
Öğrenme oranı	0.24	0.56
Momentum	0.75	0.48
Kabul edilebilir hata düzeyi	0.01	0.01
Çevrim sayısı	420	611
Min geçerlilik hatası	0.677312e-2	0.961004e-2
Min eğitim hatası	0.608348e-2	0.912152e-2
Max eğitim hatası	0.412639e-1	0.746215e-1
<u>Hata kaynakları</u>		
X1	0.297720e-2	-
X2	0.546225e-2	0.883641e-2
X3	0.681433e-2	0.817006e-2
X4	0.650612e-2	0.743571e-2
X5	0.452211e-2	0.690553e-2
X6	0.636842e-2	0.886417e-2
Doğru çözülen test örneği sayısı	183	167

Reel kodlamalı YSA, programın fonksiyon modülü ile 0-1 kodlamalı YSA ise sınıflama modülü ile çözülmüştür. Tablo 2 incelendiğinde, örnek problemlerin hepsi 6 periyotlu olmalarına rağmen, 0-1 kodlama durumunda girdi sayısının 5 olduğu görülmektedir. Bu durum bütün örnekler için sıfır başlangıç stoku varsayımından kaynaklanmaktadır. Söz konusu varsayım birinci periyotta kesinlikle o periyodun talebi kadar üretim (sipariş) yapılma zorunluluğu yaratmaktadır. Bunun sonucu olarak 0-1 kodlama durumunda x_1 değişkeni bütün örnekler için 1 değerini aldığından dolayı program bu girdi hücresini otomatikman reddetmiştir.

Oluşturulan YSA ların performanslarını değerlendirmek üzere eğitime başlamadan önce örneklerin %15 i test verisi olarak ayrılmıştır. Ağa sunulan 1275 veri ise program tarafından bir kısmı eğitim kalanı ise ezberleme kontrolü amacı ile kullanılmak üzere iki kısma ayrılmıştır. Aynı anda yapılan eğitim ve geçerlilik koşumları her bir çıktı için ayrı ayrı ve her bir ağ için ortalama bir öğrenme eğrisi üretmektedir. Ortaya çıkan 13 grafik incelendiğinde hataların belli bir çevrime kadar hızla daha sonra yavaşlayarak düştükleri ve belli bir noktadan sonra sabit kaldıkları görülmüştür. Hata dağılımlarının bu seyri her iki ağın da öğrendiğini göstermektedir. Ancak ağların ne derece iyi öğrendiklerinin değerlendirilmesi daha önce hiç görmemiş oldukları örnekler karşısındaki davranışları ile belirlenebilmektedir. Bu amaçla 225 test örneği teker teker ağlara gösterilmiştir. Bu aşamada dikkat çekici bir nokta ağların ürettiği hataların miktarlarının değil doğru

çözmüş oldukları örnek sayılarının dikkate alınmış olmasıdır. Bir diğer deyişle bir ağın bir örnek üzerinde başarılı olduğunu söyleyebilmek için optimal sonuç bulunmuş olmalıdır. Bu tercihin nedeni, her periyoda ait hataların birbirleri ile çok yüksek derecede ilişkili olmaları ve bu nedenle hata miktarlarının gerçek durumu yansıtamayacakları gerçeğidir. Tablo 1 de görüldüğü gibi reel kodlamalı YSA örneklerin yaklaşık %81 ini ve 0-1 kodlamalı YSA ise yaklaşık %74 ünü doğru çözmüştür. Bu sonuçlar her iki ağın da başarıyla eğitilmiş olduklarını ortaya koymaktadır. Sonuçlar birbirlerine oldukça yakındır ve herhangi birinin daha iyi olduğunu söylemek doğru olmayacaktır. Çünkü bir başka örnekler seti farklı bir sonuç ortaya koyabilecektir. Bu noktada önemli olan ağların eğitilebileceklerinin ortaya çıkmış olmasıdır.

4. Sonuç ve Öneriler

Bu çalışmada önceden belirli mevsimsel talep yapılarına sahip tek ürün için parti büyüklüğü problemini çözmek amacı ile geliştirilen YSA modelleri sunulmuştur. YSA modellerinin kullanımı parametrik olmama özelliklerinden dolayı giderek yaygınlaşmakla birlikte doğru olarak kurulmaları pek çok ayrıntının dikkatle ele alınmasını gerektirmektedir. Bu nedenle burada iyi bir ağın kurulması için genel aşamalar ayrıntılı bir şekilde açıklanmıştır. Birbirlerine alternatif olarak oluşturulan iki ağdan biri planlama hattının her bir periyodu için optimum üretim (sipariş) miktarlarını üretmek üzere eğitilmiştir. Diğer ağın görevi ise üretimin (siparişin) hangi periyotlarda yapılacağını belirlemektir. Ağların her ikisinde de eğitim Wagner-Whitin algoritmasından elde edilen optimum üretim (sipariş) kalıpları kullanılarak yapılmıştır. Ağ performanslarını değerlendirmek için optimal olarak çözmüş oldukları test örneklerinin yüzdesi kullanılmıştır. Yapılan analizler sonucunda ağların her ikisinin de başarıyla eğitilebildikleri ve birbirlerine çok yakın performans sergiledikleri görülmüştür. Bu durumda sonuçların yorumlanması herhangi bir dönüştürme işlemi gerektirmeyeceği için üretim (sipariş) miktarlarını belirleyen ağ diğer ağa tercih edilebilecektir.

Bu çalışmada ele alınan problemlerde gerek birim stok maliyetleri ve gerekse parti başına hazırlık (sipariş) maliyetleri planlama dönemi boyunca sabit kalmaktadır. Söz konusu maliyetlerin değişken alınması durumunda 12 adet ilave yapay sınır hücresi ağa dahil edilecektir. Ayrıca periyot sayısı daha gerçekçi olarak 12 ay veya 52 hafta olarak alınabilir. Buradakinden çok daha büyük ağlar ortaya çıkartacak olan böyle değişiklikler yer sorununun olmadığı bir çalışma için gelecek araştırma konusu olarak önerilebilir. Bir diğer araştırma konusu, çıktı olarak planlama döneminin toplam üretim (sipariş) maliyetini üreten bir YSA nın oluşturulmasıdır. Son olarak bir üretim planlama sistemi için talep tahmini, malzeme ihtiyaç planlaması ve üretim programlama faaliyetlerini bir arada ele alan bir YSA son derece işlevsel bir araç sunabilecektir.

KAYNAKÇA

- AGGERWAL, A. and PARK, J.K., "Improved Algorithms for Economic Lot Size Problems", **Operations Research**, 1993, 41 (3), s. 549-571.
- ARBIB, M.A., **Brains, Machines, and Mathematics**, Newyork, 1987.
- BAHL, H.C., RITZMAN, L.P., GUPTA, J.N. "Determining Lot Sizes and Resource Requirements: a Review", **Operations Research**, 1987, 35 (3), s.329-345.
- BLACKBURN, J.D. and MILLEN, R.A., "Heuristic Lot Sizing Performance in a Rolling Schedule", **Environment. Decision Sciences**, 1980, 11 (3), s. 691-701.
- BRACE, M.C, BUI-NGUYEN, V., SCHMIDT, J., "Another Look at Forecast Accuracy of Neural Networks", **In: Proceedings of the Second Joint Forum on Applications of Neural Networks to Power Systems**, Yokohama, Japan, s. 19-22, 1993.
- CARLSON, R., JUCKER, J.V., KROPP, D., "Less Nervous MRP Systems: a Dynamic Economic Lot Sizing Approach", **Management Science**, 1979, 25 (8), s. 754-61.
- CHOUYEKI, M.H., "A Designed Experiment on the Use of Neural Network Models in Short Term Hourly Load Fore Casting",. PhD Dissertation, Department of Industrial and Systems Engineering, The Ohio State University, Ohio, 1995.
- COLEMAN, B.J., MCHNEW, M.A., "A Technique for Order Placement and Sizing", **Journal of Purchase and Materials Management**, 1990, 26(1),s. 32-40.
- CONNOR, J., ATLAS, L., "Recurrent Neural Networks and Time Series Prediction", **IJCNN-91 Seattle, International Joint Conference**, s. 301-306., 1991.
- DEMATTEIS, J.J., "An Economic Lot Sizing Technique I: The Part Period Algorithm", **IBM System Journal**, 7(1), 1968, s. 39-48.
- DEMATTEIS, J.J., Mendoza, A.G. "An Economic Lot Sizing Technique", **IBM System Journal**, 1968, 7, s. 30-34.
- EMILE, F. and BEALE, R., **Handbook of Neural Computation: Supplement 1**, CRC Pres, 1997.
- EZZIANE, Z.H., MAZOUZ, A.K., HAN, C., "Neural Network Approach for Inventory Control", **Intelligent Robotics and Computer Vision XI. SPIE**, 1992, s. 501-9.
- GAAFAR, L.K. and CHOUYEKI, M.H., "A Neural Network Model for Solving The Lot Sizing Problem", **Omega**, 28, 2000, s. 175-184.
- HECHT-NIELSEN, R., **Neurocomputing. Reading**, Massachusetts, Addison-Wesley, 1990.

- HINTON, G.E. "Connectionist Learning Procedures", **Artificial Intelligence**, 1989, 40, s. 185-234.
- HOPTROF, R.G., BRAMSON, M.J., Hall, T.J., "Forecasting Economic Turning Points with Neural Nets", **IEEE xplore**, 1991, 347-352.
- HORNIK, K., "Some New Results on Neural Network Approximation", **Neural Networks**, 1993, 6, s. 1069-1072.
- JADID, M.N. and FAIRBAIRN, D.R., "Predicting Moment Curvature Parameters from Experimental Data", **Engineering Application Artificial Intelligence**, 1996, 9 (3), s. 303-319.
- KIM, S., LEE, Y., AGNIHOTRI, D., "A Hybrid Approach to Sequencing Jobs Using Heuristic Rules and Neural Networks", **Production Planning and Control**, 1995, 6(5), s. 445-454.
- KROPP, D., CARLSON, R., Jucker, J.V., "Heuristic Lot Sizing Approaches for Dealing with MRP System Nervousness", **Decision Science**, 1983, 14 (2), s. 156-186.
- LAARHOVEN, P.J., AARTS, E.H, LENSTRA, J.K., "Job Shop Scheduling by Simulated Annealing", **Operations Research**, 1991, 40, s. 113-125.
- LACHTERMACHER, G. and FULLER, J.D., "Backpropagation in Time Series Forecasting", **Journal of Forecasting**, 1995, 14, s. 381-393.
- LEE, Y.Y., KRAMER, B.A., HWANG, C.L., "A Comparative Study of Three Lot Sizing Methods for the Case of Fuzzy Demand", **International Journal of Operations & Production Management**, 1991, 11 (7), s. 72-80.
- MASTERS, T., **Practical Neural Networks Recipes in C++**, Academic Press, San Diego, 1993.
- NTUEN, CA. "A Neural Network Model for a Holistic Inventory System", **International Industrial Engineering Conference**, s. 435-444, 1991.
- PAO, Y.H., **Adaptive Pattern Recognition and Neural Networks**, Addison-Wesley, 1989.
- PATTERSON, J.W., LaForge, R.L., "The Incremental Part Period Algorithm: an Alternative to EOQ", **Journal of Purchase and Material Management**, 1985, 21 (2), s. 28-33.
- RADZI, N. H. M., HARON, H., Irdawati, T., JOHARI, T., "Lot Sizing Using Neural Network Approach", **Regional Conference on Mathematics, Statistics and Applications**, Malaysia, ss. 1-8, 2006.

-
- SIKORA, R., CHHAJED, D., SHAW, M., "Integrating the Lot Sizing and Sequencing Decisions for Scheduling a Capacitated Flow Line", **Computers & Industrial Engineering**, 1996, 30, s. 659-679.
- SILVER, E.A. and MEAL, H.C., "A Heuristic for Selecting Lot Size Quantities for the Case of Deterministic Time-Varying Demand Rate and Discrete Opportunities for Replenishment", **Production Inventory Management**, 1973, 14 (2), s. 64-74.
- SILVER, E.A., MILTENBURG, G.J., "Two Modifications of the Silver-Meal Lot Sizing Heuristic", **INFOR**, 1984, 22 (1), s. 56-69.
- STEHOUWER, H.P., AARTS, E.H.L., WESSELS, J. "Multi Layered Perceptrons for on-line Lot Sizing", Technical Report, The Netherlands: Eindhoven University of Technology, ss. 279-287, 1995.
- UPADHAYA, B. and ERYUREKA, E., "Application of Neural Networks for Sensory Validation and Plant Monitoring", **Neural Technology**, 1992, 97, s. 170-176.
- WAGNER, H.M. and WHITIN, T.M., "Dynamic Version of the Economic Lot Size Model", **Management Science**, 2004, 50(12), s. 1770-1774
- WHITIN, T. M., **The Theory of Inventory Management**, 2nd ed., Princeton University Press, Princeton, NJ., 1957.
- YIH, Y., LIANG, T., MOSKOWITZ, H., "Robot Scheduling in a Circuit Board Production Line: a Hybrid OR/ANN Approach", **IIE Transactions**, 1993, 25, s. 26-33.
- ZHIWEI, Z., HEADY, R.B., LEE, J., "A Simple Procedure for Solving Single Level Lot Sizing Problems", **Computers & Industrial Engineering**, 1994, 26 (1), s. 125-131.
- ZWIETERING, P.J., "The Complexity of Multi-Layered Perceptrons", Ph.D. thesis, Eindhoven University of Technology, 1994.
- ZWIETERING, P.J., KRAAIJ, M.J.A.L., AARTS, E.H.L., WESSELS, J., "Neural Networks and Production Planning", **In Proceedings of the Fourth International Conference on Neural Networks and their Applications**, s. 529-542, 1991.