

### Özet

Günümüzde salt bilgisayar destekli öğretim yerini zeki öğretim sistemlerine bırakmaktadır. Bugüne değin bu alanda pek çok çalışma yapılmış olmasına karşın, günün değişen gereksinimleri yeni çalışmaların ve bu çalışmaların günlük hayata aktarılmasını zorunlu kılmaktadır. Temel ZÖS mimarileri de söz konusu değişimlerden payını almakla beraber genel esasları itibarıyla şu an için belirgindir. Uzman sistemlerin yaygınlaşması ZÖS'nin gelişiminde yararlı olacaktır.

### Abstract

Today, intelligent tutoring systems takes place of complete computer aided instruction. Although there are many studies in this field, new research and practice must be done as the needs for today are changing. While basic ITS architecture is also influenced by these changes, its general components are determined at least for now. The development of expert systems will be useful for the development of ITS.

### Giriş

Bilgisayarlar, 20 yılı aşkın bir süredir dersliklerin bir parçası olmuştur. 1976 da mikrobilgisayarların hayatımıza girmesiyle etkili olmaya başlamışlardır. Bilgisayar, bireylerle hızla etkileşime girmeyi, çeşitli biçimlerdeki çok sayıda bilgiyi saklayıp işlemeyi ve geniş bir dizi görsel-işitsel girdiyi göstermek için diğer medya araçlarıyla birlikte kullanmayı sağlayabilmektedir. Bilgisayar bu özellikleriyle öğretimde potansiyelini de ortaya koymaktadır. Çeşitli öğretim etkinliklerinde bilgisayarın kullanılması giderek yaygınlaşmaktadır[1]. Bilgisayarlar bir eğitim unsuru olarak hayatımızdaki yerini almaktadırlar çünkü:

- Çocuklar tarafından kontrol edilebilen grafiksel sunular sağlayarak onları motive edebilirler,
- Hızlı bir şekilde doküman sunabilirler,
- Bireysel eğitim sağlayabilirler,
- Anında hata tespiti ve geri besleme imkanı sunabilirler,
- Öğretmene, öğrenciyle fert bazında veya küçük gruplar halinde çalışma serbestisi verirler,
- Öğretmeni, hazırlayacağı raporlar için öğrenciler hakkında bilgi edinmek, sınav sonuçlarını değerlendirmek ve her öğrencinin gelişimini takip etmek gibi idari ve eğitsel faaliyetlerden kurtarabilirler.

### Amaç

Bilgisayarın eğitimdeki başlıca kullanım alanı pratik yapılması yoluyla destek vermektir ve genellikle Bilgisayar Destekli Öğretim (BDÖ) olarak tanımlanır. Bu tür yazılımların hazırlanması öğrenci-bilgisayar diyalogu içeren konu anlatımı veya simülasyon yazılımlarından daha kolaydır. Bu yazılımlar matematik ve yabancı dil kelime ve dilbilgisi konularında etkili olmuşsa da aynı metot daha karmaşık konuların sunulmasında başarısız olmuştur. Öğrenciler sunuların sıkıcı olmaları nedeniyle tatmin olmamışlardır. "Alıştırma ve Uygulama" terimleri tekrar ve esnek olmamakla aynı anlama gelmeye başlamıştır. Üstelik alıştırma ve uygulama olarak sınıflandırıldığı zaman, eğitim materyallerinin şüpheyle karşılandığı bir gerçektir.

Oyun ve simülasyonlara dışarıdan bakıldığında eğitimi yapılacak olan konunun yarışmanın bir parçası haline geldiği görülür. Esnekliği olmayan yordamlar öğrencilerin görecekları konuya ilişkin yapabilecekleri seçimlere dönüşür. “Tekrar” geniş bir veri tabanından elde edilen muhteviyat sayesinde gizlenmiş olur (Tekrar edilmesine rağmen farklı seçenekler öğrenciye tekrar değilmiş gibi gelir). Tekrar edilecek olmasına rağmen ana konular, onları daha çekici yapan yeni ve farklı bölümlerde yer alacaklardır.

Gelişmekte olan ülkelerde, eğitsel yazılım daha fazla yaygın olduğuna göre sözlü geleneklerden, yazılı kayıtların zorunlu olduğu çağdaş kültürlerle bir değişim önerdiğinden alıştırma ve uygulamalara canlandırılmış bir ilgi olacaktır. Sonuçta okuma yazma oranı artan gelişmiş ülkelerde bu metodolojiyi incelemede canlandırılmış bir ilgi vardır. Bireyselleşmiş eğitimi sağlamak için bilgisayarın kullanımı okuma yazma bilmeyen yetişkinleri motive edebilir.

Bununla birlikte eğitimde bilgisayarın kullanımını yaygınlaştırmak için daha yenilikçi ve “zeki” yazılımlara ihtiyacımız vardır. “Zeki” demekle şu tip yazılımı kastediyoruz (diğer olanakları dahil):

- Alan bilgisini, kontrol (pedagojik) bilgisinden ayıran böylece sistemin tasarımcı tarafından ön-kodlanmasına değil özel akışlar sunmasına izin veren,
- Öğrencilere, programın kendilerine gösterdiği materyaller üzerinde daha fazla kontrol sağlayan,
- Öğrencilere, yaptıkları hatalara göre acil geribesleme sağlayan (isteğe bağlı),
- Öğrencilere içinden çıkamayacakları durumlarda, öğretmenlik ya da çalıştırıcılık sağlayan,
- Öğrencilere, aksiliklerle karşılaşacakları durumlarda kendilerine yardım etmesi için seçebilecekleri, isteğe bağlı seçeneklere sahip, öğretmene başvurmak üzere tavsiyeler de içeren bir “Yardım” menüsü içeren bir yazılım

Bu açıları eğitsel yazılım içerisine katmak için çalışan araştırmalar ile sık sık hatırlanan terim Zeki Bilgisayar Destekli Öğretim (ZBDÖ) ya da sonraları daha başarılı şekilde ortaya konulan Zeki Öğretim Sistemleri (ZÖS) terimidir [2].

Eğitim ve eğitim psikolojisi, pedagojik uzmanlığın parçalarını tamamlama, deneyime dayalı kullanışlı teknikler edinme ve kullanışlı eğitsel araçlar üretme gayretleri dolayısıyla, birbiriyle sıkı sıkıya ilişkili araştırma alanlarıdır. Bilgi iletişim sistemleri için hazırlanan, dolaysız pedagojik uzmanlık bilgisayar modeller, muhtemelen, eğitim araştırmalarına hem teorik ve hem de pratik katkıları olacaktır. Gerçekte, eğitim toplumu ile ZÖS araştırmacıları arasındaki bağ her şeye rağmen zayıf olmuştur. Bu durum hızla değişmektedir [3].

## **ZÖS Mimarileri**

Zeki Öğretim Sistemleri’nin kısa ve açık bir tanımı “neyi öğreteceğini, kime öğreteceğini ve nasıl öğreteceğini bilen, yapay zeka ortak oluşumunda yer alan tekniklerden yararlanarak tasarlanmış bilgisayar programları” şeklinde yapılabilir [4].

ZÖS pek çok yapıda olabilmektedirler; ancak esasen bir öğretim sisteminin temel bileşenlerini, öğrenim ortamında gerek öğrenci ve gerekse sistemin esnekliğine izin veren ve böylelikle gerçekte öğrenci ve öğretmenin birebir oturup öğrenme ve öğretme girişimini taklit eden bir yöntemle ayrılmaktadırlar.

Varolan ZÖS’nde kullanılan mimarilerin sayısı ve farklılıkları bu alanda yapılan çalışmaların doğasının deneysel olmasına bağlı olarak şartıcı derecede fazladır. Halen, literatürde ZÖS, üzerinde bir anlaşmaya varılmış olan dört temel modül içermektedirler [4]:

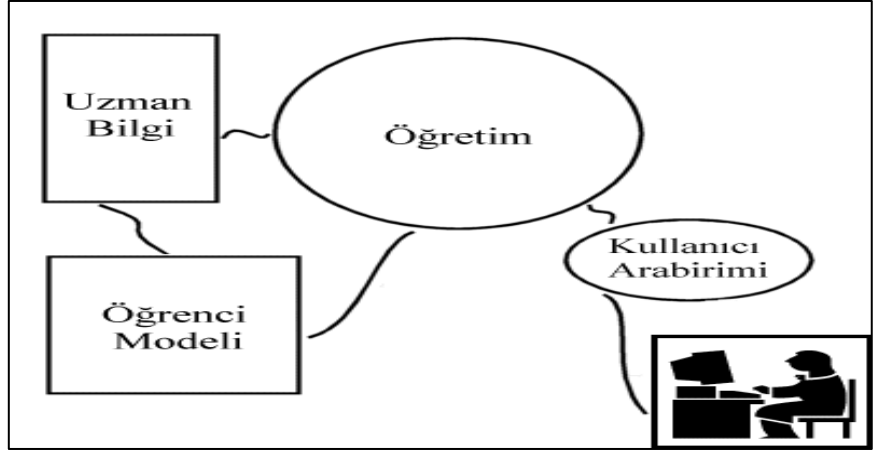
- Uzman Bilgi Modülü
- Öğrenci Modeli Modülü
- Öğretim Modülü
- Kullanıcı Arabirimi Modülü

ZÖS'nin genel yapısı Şekil 1'de verilmiştir. Bu şekil genel bir gösterimdir ve özel herhangi bir sistemi tanımlamamaktadır. Uygulamada hafif varyasyonlar göstermekle beraber genel yapıya tam olarak oturan veya tamamen farklı olan sistemler de bulunmaktadır.

Buradaki modüllerin işlevlerine göz atmak istediğimizde karşımıza şu özellikler çıkmaktadır:

**UZMANLIK MODÜLÜ:** Sistemin öğretmeye çalıştığı alan bilgisi bu modül içerisinde saklanır. Dahası, öğrenciye kazandırılması arzulanan yetenek ve içeriklerin gerçek modeli, özgün alana ilişkin uzmanlığın dinamik bir formu ile donatılan bu modül içerisinde oluşturulmaktadır. Bu bileşen tarafından gerçekleştirilen iki ana işlev vardır [5]:

1. Soru, cevap ve açıklamaların geliştirilmesi ve böylelikle bir kaynak gibi davranım,
2. Kavrama düzeyini saptamak için standartları belirlemek açısından kullanıcı performansının değerlendirilmesi.



Şekil 1

### Genel Olarak ZÖS Mimarisinde Yer Alan Modüller [3]

Bu modül öğrencinin performansının değerlendirilmesi ve öğretim içeriğinin oluşturulması görevlendirildiğinden, kolay kullanım adına modülün alan bilgisinin bir bilgisayar programı çatısı altında organize edilmesi gereksinimi vardır. Bu organizasyon çoğunlukla zaman alıcı, karmaşıktır ve buna bağlı olarak organizasyon için metotların araştırılması ve bilginin sunulması bir uzman modülünün geliştirilmesinde önemli bir noktadır [4].

**ÖĞRENCİ MODELİ MODÜLÜ:** Öğrencinin bilgi düzeyini ölçmek için ve ayrıca onun kavrayışını tahmin etmeye çalışmak için bir öğrenci modeli ve öğrencinin o anki bilgi düzeyine erişmek için uslaamlama (muahakeme) metotları kullanılmaktadır. Bu, aynı işlem üzerinde bilgisayar tabanlı uzmanın davranışıyla öğrencinin performansının kıyaslanmasıyla yapılmaktadır. Öğrencinin bilgisinin modellenmesi ve öğrenme davranışı temelde iki yordam kullanılmaktadır [5]:

1. Öğrencinin uzmanlaştığı ya da öğrenmeye kalkıştığı alanlar olan bilgi yapısı ağı içerisinde planlandırılmak. Diğer bir deyişle, öğrencinin bilgi düzeyi, uzman ve sonuçta oluşan bir altkümeneinkinlikle kıyaslanması ustalık derecesinin ölçülmesi için kullanılmaktadır. Bu yordama katmanlama denmektedir. Öğrencinin davranışı, sistemin basit olarak bilgi tabanı içerisinde var olan doğru ya da yanlış bilgi kalıpları ile tanımlanmaktadır.

2. Öğrencinin yeteneği anlama ve muhakeme işlemi hakkında sonuç çıkartmak için öğrencinin tepki geçmişine kalıpsal tanımlama uygulanması tepkilerin elde edilmesi için kullanılırdı. Bu aşama aynı zamanda *tanımlama* olarak da adlandırılır ve istatistiksel analiz gibi metotlarda kullanabilmektedir.

İdeal koşullarda öğrenci modeli öğrencinin performans ve öğrenimini etkileyebilecek tüm davranış ve bilgilerini kapsamalıdır, ancak gerçekte, böyle bir model oluşturmak insan davranışının tüm duyuların, görüşün, sesin ve hatta yüz hareketlerinin bir bileşeni olan karmaşık bir konu olmasından dolayı neredeyse imkansızdır. Bu yüzden, bir ZÖS’inde klavye tek iletişim aracı olduğundan, insan öğretmenlerin düşünülmesi tanımlama yeteneklerinin çoğundan eksik kalmaktadır.

Genellikle bir öğrenci modelinin kullanım işlevleri şu şekildedir [4]:

1. Düzeltici: Öğrencinin bilgisindeki hataların giderilmesi için bir rehber olarak,
2. Detaylı: Öğrencinin bilgisini tamamlamada bir rehber olarak,
3. Stratejik: (1) ve (2)inci maddelerde sözü edilenlerden farklı olarak öğretim stratejisindeki ana sapmaların düzenlenmesinde yardımcı olarak,
4. Teşhis: Öğrencinin bilgisinde yer alan hataların belirlenmesinde yardımcı olmak amacıyla,
5. Tahmin: Öğrencinin öğretim eylemine karşı olası tepkilerinin belirlenmesine yardımcı olmak amacıyla,
6. Değerlendirme: Öğrencinin ya da ZÖS’nin değerlendirilmesinde yardımcı olmak amacıyla.

**ÖĞRETİM MODÜLÜ:** Öğretim modülü, metot, sunu zamanlaması ve sistemin içerisinde barındırması gereken öğretim materyallerinin bir kümesinden oluşmaktadır. Genellikle halen varolan ZÖS’nde iki sunu metodu bulunmaktadır: *Sokralsal ve çalıştırıcı* metotlar. Birinci metot, kendi yanlış anlamalarını giderme işlemi için öğrenciye yöneltilen bir soru setini devreye sokmaktadır. Diğeri ise, öte yandan, öğrenci için bilgisayar oyunları gibi eğlenceli bir ortam yaratmaktadır ve böylelikle ilişkili yetenekleri ve genel problem-çözümü alışkanlıklarını öğrenebilmektedirler.

#### **Uygulama Örnekleri**

**SCHOLAR:** Carbonell’in 1970 yılındaki bir çalışması olan zeki öğretim alanında ilk öncü sayılabilecek programdır [5] [6]. SCHOLAR’a dahil edilen bazı önemli özellikler şunlardır:

- Olguların, içeriklerin ve yordamların oluşturduğu bir ağ yapısı içerisinde karmaşık ancak iyi tanımlanmış bir veritabanı yapısı,
- Bilgi-tanımlama kelimeleri içeren bir ağ ve çok-dallı ağaç yapısında olaylar,
- Sokratsal öğretim diyalogu,
- Cevaplar için sonuç çıkarım stratejileri anlamsal ağın içeriğinden bağımsızdır.

SCHOLAR, öğrencilere *Güney Amerika coğrafyası* ’nı öğretmeyi amaçlayan kimi zaman öğrencinin programa kimi zaman ise programın öğrenciye sorular sorduğu bir zeki öğretim sistemiydi. SCHOLAR’ın uzman bilgisi modülü anlamsal bir ağ formunda Güney Amerika coğrafyasını barındırıyordu. Bu yapı sistemi bellek problemlerinden, sorulara izin vermeyi cevapların saklanmasına gerek duyulmadan sağlayan yapısından dolayı kurtarmaktaydı. Bu devrimsel gelişimlerin yanı sıra, basit olarak ZÖS alanında ilk sistemlerden biri olduğu için SCHOLAR’ın belirli bir takım dezavantajları ya da güçsüz yönleri de vardı. Bu dezavantajlar kendini öğretim stratejisinin ve dil işleme yeteneğinin kısıtlı olmasında göstermekteydi. SCHOLAR’ın eksikleri olsa da ZÖS dizaynı ve uygulamaları için yaşamsal pek çok içeriği getirmişti.

**BUGGY:** ‘Hata’ yapan öğrenci uyarlaması yaparak ‘öğretmen’ öğrencilerin söz konusu hatayı bulmalarını sağlayan bir matematik oyunu programıydı [7]. Sistemli hatalar yapan program çerçevesinde, öğretmen öğrencilerin çoğu özel ve önemli bir hatayı bulduklarını düşünmekteydiler.

**PROUST:** Programcılığa yeni başlayanlar tarafından yazılan Pascal programlarındaki yazımsal olmayan hataları bulmak üzere tasarlanan, bilgi tabanlı bir sistemdir[8]. Her çeşit hatayı bulur. Buna ek olarak, hataların nasıl ortadan kaldırılacağını belirler. PROUST,

sentezle-analiz yaklaşımını kullanarak programları analiz eder; daha önce belirlenmiş olan program gereksinimlerini, söz konusu gereksinimleri yerine getirebilecek metodları bulmak için inceler. Daha sonra sistem, mümkün olan her metodu programcının metoduyla karşılaştırır ve böylece de programcılık bilgisine gereksinim gösterir.

PROUST, bir ZÖS olarak, bir hayli başarılıydı ve tecrübesiz programcılarca yazılmış olan kod içerisindeki pek çok hatayı bulmayı becermiştir. PROUST üzerine kurulabilecek diğer bir mantıklı adım da, sadece öğrencinin hatalarını düzelteren değil de, aynı zamanda gereksinim duyulduğunda öğrenciye pratik kazandırmak için gerekli örnekleri sağlayabilecek bir otomatikleştirilmiş programlama kursu olmasıdır.

WEST: Öğrencilere “How the West was won?” (“Batı nasıl kazanıldı?”) adı verilen, bir oyun oynatılarak öğretimin gerçekleştirilmesini amaçlayan bir çalıştırıcı programın ilk örneklerinden biridir [4] [6]. Oyun, her bir oyuncu tarafından üç anahtar sayı üzerinde dört işlemden yararlanılarak değişiklik yapılmasına dayanmaktaydı. WEST, ZÖS’in farklı modüllerinin çok farklı sistemlerde kullanılabilmesini mükemmel bir şekilde vurgulayan bir programdı. Bu yüzden, WEST ortaya koyduğu resmi olmayan öğretim atmosferine bağlı olarak bir öğreticiden çok bir “*çalıştırıcı*” olarak adlandırılmaktadır.

### **Tartışma**

Klasik bilgisayar destekli öğretim, öğretim araç gereçleri içerisinde bilgisayarı dahil etmeyi öngörürken, ZÖS’nden yararlanılarak hazırlanan eğitim programlarıyla öğreticinin yanında bilgisayar bir destek elemanı olarak bulunabilmekte, hatta öğreticinin bulunmadığı ortamlarda öğreticinin kısmen yerini alabilmektedir. Bu görüşün en iyi uygulaması ancak uzman sistemlerin yeterince geliştirilmesiyle ortaya çıkmaktadır.

ZÖS’ nin eğitimde uygulama alanları şu şekilde karşımıza çıkmaktadır:

1. Bireye özgün öğretim ortamı sağlanabilmesiyle,
2. Genel alıştırma, uygulamalarla,
3. Öğrenci kontrolü istenildiğinde,
4. İstatistiksel veriler talep edildiğinde,
5. Simülasyonlarla ekonomi sağlanması arzulanığında,
6. Öğretim programına ilgi çekilmesinin sağlanması için oyunlarla,
7. Hayal gücünün ve yaratıcılığın geliştirilmesi beklendiğinde,
8. Problem çözümünün sembolizasyonu istendiğinde.

Bu örnekler gereksinimlere bağlı olarak kolaylıkla artırılabilir.

Buradan da kolayca görüleceği üzere ZÖS’ nin hayatımıza girmesiyle eğitim alanında yeni ufuklar açılacaktır. Klasik öğretim sistemleriyle elde edilebilecek en yüksek başarı oranlarının ZÖS devreye sokulduğunda ne şekilde artacağı daha ileri çalışmaları gerektirmektedir.

### **Kaynakça**

- [1] Kaya, Z.(1999). “Bilgisayar Destekli Öğretim ve Ergonomi” *Birinci Uluslar arası Katılımlı Bilgi Teknolojileri Sempozyumu Bildirileri*. Bursa: Uludağ Üniversitesi Eğitim Fakültesi.
- [2] Sleeman, D. & Brown, J. S. (1982). *Intelligent tutoring systems*. London ; New York : *Academic Press*. 345 p.
- [3] Wenger, E. (1987). *Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge*. Los Altos, Calif. : *Morgan Kaufmann Publishers*. 486 p.
- [4] Nwana, S. N. (1990). *Intelligent Tutoring Systems: an Overview*. *Artificial Intelligence Review*, (4), 251-277.

- [5] Fischetti, E. & Gisolfi, A. (1990). From Computer-Aided Instruction to Intelligent Tutoring Systems, *Educational Technology*. August 1990, 7-17.
- [6] O'Shea, T. & Self J. (1983). Learning and Teaching with Computers: Artificial Intelligence in Education. *The Harvester Press, Brighton, Sussex*, 307 p.
- [7] Burton, R. R. & Brown, J. S. (1979). An investigation of computer coaching for informal learning activities. *Int. J. Man-Mach. Stud.*, 11, 5-24.
- [8] Kearsley, G. (1987). Artificial Intelligence and Instruction: Applications and Methods. *Addison-Wesley Publishing Company, Massachusetts*, 1987.