GU J Sci, Part C, 5(4): 27-34 (2017)

Gazi Üniversitesi

Fen Bilimleri Dergisi

PART C: TASARIM VE TEKNOLOJİ

dergipark.gov.tr/http-gujsc-gazi-edu-tr

Genetik Algoritma ile Akıllı Test Sayfası Oluşturma

[1]Ufuk TÜL, [2]Adem TUNCER

[1] *Yalova Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği ABD, Yalova*

[2] *Yalova Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Yalova*

Makale Bilgisi

Başvuru: 05/10/2017
Düzeltme: 20/11/2017
Kabul: 20/11/2017

Anahtar Kelimeler

Genetik algoritma
Soru bankası
Test sayfası oluşturma
Kullanıcı arayüzü
Web tabanlı uygulama

Öz

Bu çalışmada, bir soru bankası içerisinden istenen ölçütlere bağlı olarak akıllı test sayfası oluşturma probleminin genetik algoritma ile çözümü sunulmuştur. Akıllı test sayfası oluşturma, soru bankasındaki her bir sorunun pek çok özniteliğe sahip olmasına bağlı olarak çok parametreli bir optimizasyon problemi olarak ele alınmaktadır. Genetik algoritma optimizasyon problemlerinin çözümünde sıkça kullanılan paralel arama özelliğine sahip sezgisel arama algoritmasıdır. Çalışmada standart genetik algoritmanın çaprazlama ve mutasyon operatörlerinde yapılan değişiklikler ile genetik algoritmanın performansının artması ve istenen kalitede test sayfalarının oluşturulması sağlanmıştır. Deneysel sonuçlar, iyileştirilmiş genetik algoritmanın aynı koşullardaki standart genetik algoritma ile karşılaştırıldığında daha etkili olduğunu göstermektedir. Yapılan çalışmada, kullanıcıların genetik algoritma ve test sayfası için istediği ölçütleri belirleyebildiği ve algoritmayı çalıştırabildiği web tabanlı bir kullanıcı arayüzü uygulaması geliştirilmiştir.

Keywords

Genetic algorithm
Question bank
Test page generation
User interface
Web-based application

Intelligent Test Paper Generation with Genetic Algorithm

Abstract

In this study, the solution of the problem of generating an intelligent test paper with a genetic algorithm is presented depending on the required criteria in a question bank. Generating the intelligent test paper is considered as a multi-parameter optimization problem, depending on whether each question in the question bank has many attributes. A genetic algorithm is a heuristic search algorithm with parallel search feature which is often used to solve optimization problems. In the study, the changes in the crossover and mutation operators of the standard genetic algorithm increased the performance of the genetic algorithm and created the test papers in the required quality. Experimental results show that the improved genetic algorithm is more effective when compared to the standard genetic algorithm in the same conditions. In the study, a web-based user interface application was developed in which users can set the criteria for genetic algorithm and test paper and can run the algorithm.

1. GİRİŞ (INTRODUCTION)

Eğitim öğretim hayatında önemli bir rol oynayan sınav sisteminde kâğıtsız otomasyon sistemleri toplumun her kesimini kapsayacak şekilde giderek yaygınlaşmaktadır. Bu sistemlerde ihtiyaç duyulabilen akıllı test sayfası oluşturma, testi uygulamak için önemli adımlardan biridir. Otomatik olarak oluşturulan akıllı test sayfasındaki temel yaklaşımla, çoklu kısıtlar göz önüne alınarak istenilen yüksek kalitede test sayfaları oluşturulabilmektedir. Sistem, sınav sisteminin reformunda önemli bir rol oynamakta ve test sorusu hazırlayan uzmanların çalışmalarını büyük ölçüde kolaylaştırabilmekte ve azaltabilmektedir.

Test sayfası oluşturma problemi, bir çok kısıt altında çok parametreli bir optimizasyon problemidir. Bu problemin çözümüne ilişkin farklı algoritma ve yöntemler kullanılmaktadır. Geleneksel algoritmalar ile test sayfası oluşturma yöntemleri, düşük başarı oranı, tekrarlı ve istenen kalitede soru üretilememesi gibi çeşitli dezavantajlara sahip olabilmektedir [1]. Test sayfası üretimi konusunda rastgele seçim (random select) ve geri izleme (backtracking) gibi bazı algoritmalar denenmiş olsa da bu algoritmalar akıllı olmamaları, fazla sayıda kısıtlar olduğunda tatmin edici sonuç verememeleri, fazla zaman almaları gibi

*İletişim yazarı, e-mail: ufuktul@gmail.com

DOI: 10.29109/http-gujsc-gazi-edu-tr.341977

çeşitli sebeplerle başarısız olmuşlardır. Akıllı test sayfası üretiminde başarıyı sağlamak için geleneksel algoritmaların yanı sıra sezgisel yöntemler de kullanılmaktadır. Literatürde en fazla kullanılan sezgisel yöntemlerden biri olan genetik algoritma (GA) paralel olarak çalışabilme, akıllı arama, sağlamlık, basit yapısı gibi özellikleriyle birlikte doğal ve evrimsel bir benzetim modeli sunmaktadır [2].

Matematiksel olarak ifade edilemeyen ve çözüm için belirli bir fonksiyona sahip olmayan karmaşık problemlerin çözülmesinde GA kullanımı, başarı oranını arttırmakta ve problemin daha hızlı bir şekilde çözülebilmesine katkı sağlamaktadır. Test sayfası oluşturma probleminin çözümünde diğer algoritmalara kıyasla GA gibi sezgisel algoritmalar birtakım dezavantajların önüne geçebilmektedir. Ancak standart GA'nın da bazı durumlarda yetersiz kaldığı görülmektedir. Standart GA, kısıtlı sayıda soru sayısı ve çözümü etkileyecek fazla sayıda kısıtlama kullanımında yeterli başarı oranını sağlayamamaktadır. Bunun yanında tekrarlı soru üretimi gibi istenmeyen bazı durumların çözümünde de standart GA'nın başarısız kaldığı durumlar söz konusu olabilmektedir. Bu çalışmada, test sayfası probleminin çözümü için, bazı yönleri geliştirilmiş bir GA ile daha kaliteli, etkin ve başarılı bir akılı test sayfası oluşturulması sağlanmıştır. Standart GA'da kullanılan çaprazlama operatöründe değişiklik yapılarak bölüm tabanlı bir çaprazlama işlemi yapılmış ve çaprazlama işleminden sonra da bölümler için istenilen soru sayılarının aynı kalması sağlanmıştır. Mutasyon operatörü için de yapılan değişiklikle, mutasyona uğrayacak olan sorunun, bölüm bilgisine göre ve tekrarsız olacak şekilde rastgele seçilmesi sağlanmıştır. Test sorularının oluşturulmasında, zorluk seviyesi, soru puanı, bilgi puanı, sorunun seçilme sıklığı ve sorunun cevaplama süresi ölçüt olarak kullanılmıştır. Literatürdeki çalışmalarla kıyaslandığında pek çok çalışmaya göre daha fazla ölçüt dikkate alınmış ve ölçütleri kullanıcı tarafından belirlenen ağırlık değerleriyle birlikte daha esnek bir test oluşturma çalışması sunulmuştur.

2. LİTERATÜR TARAMASI (LITERATURE REVIEW)

Xiumin ve diğ. [1] tarafından gerçekleştirilen çalışmada, test sayfası oluşturma işlemi için GA kullanılmış ve GA'daki başlangıç nüfusunun rastgele olmayan optimize edilmiş bir nüfus olması önerilmiştir. Adaptif bir çaprazlama ve mutasyon yöntemi kullanılarak bir önceki nüfusta ortalama uygunluk değeri ile yeni nüfustaki ortalama uygunluk değerleri karşılaştırılmış ve buna göre çaprazlama ve mutasyon işlemlerinin uygulanıp uygulanmayacağına karar verilmiştir.

Nie Jun [2] tarafından gerçekleştirilen çalışmada, GA kullanılarak bölüm tabanlı çaprazlama işleminin uygulandığı, çaprazlama ve mutasyon işlemlerinin de daha önce belirlenen bir olasılık formüllerine göre gerçekleştirildiği belirtilmiştir. Aday test sayfasının özelliklerine göre adaptif bir formül uygulanmış ve bu formüle göre çaprazlama işleminin yapılıp yapılmayacağı belirlenmiştir. Amaç fonksiyonu içinde soruların zorluk seviyeleri, puanı, bilgi seviyesi puanı ve farklılık puanı gibi parametreler yer almaktadır. Standart GA ile karşılaştırıldığında daha başarılı sonuçların elde edildiği belirtilmiştir.

Zhang ve Zhu [3] tarafından gerçekleştirilen çalışmada, GA'nın çaprazlama işlemi için test sayfasının soru tiplerine göre bölümlere ayrıldığı ve bu bölümler için ayrı ayrı çaprazlama işleminin yapıldığı belirtilmiştir. Hedeflenen çaprazlama işlemi ile standart GA'ya göre daha başarılı sonuçların elde edildiği vurgulanmıştır. Çalışmada amaç fonksiyonu için soruların zorluk seviyeleri ve puanları kullanılmıştır.

Xiao-Ying Sun [4] tarafından gerçekleştirilen çalışmada, parçacık sürü optimizasyonu algoritması kullanılarak test sayfası oluşturma problemine çözüm aranmıştır.

Ying Shan [5] tarafından gerçekleştirilen çalışmada, test sayfası oluşturma problemi için GA'da çoklu iş parçacığı yönteminin kullanıldığı belirtilmiştir. Benzetim sonuçları standart GA ile karşılaştırıldığında, önerilen yaklaşımın daha hızlı ve iyi sonuç verdiği belirtilmiştir.

Liron ve Jianwei [6] tarafından gerçekleştirilen çalışmada, test sayfası oluşturma problemine çözüm için GA'nın çaprazlama ve mutasyon operatörleri üzerinde iyileştirmeler yapıldığı belirtilmiştir. Standart çaprazlama işleminden farklı olarak, çaprazlama işleminden sonra yeni bireylerin amaç fonksiyon değerlerinin hesaplanıp, ebeveyn bireylerin amaç fonksiyon değerleri ile karşılaştırıldığı ve karşılaştırma sonunda çocuk bireylerin değerleri ebeveyn bireyler ile aynı ise çocuk bireylerin elendiği bir çalışma yapıldığı belirtilmiştir.

3. TEST SAYFASI MATEMATİKSEL MODELİ (THE MATHEMATICAL MODEL OF TEST PAGE)

Bilgisayar teknolojisi alanındaki geniş öğrenme uygulamaları ile birlikte bilgisayar destekli testlerin hazırlanması gün geçtikçe artmaktadır. Kâğıtsız sınav sistemlerinin giderek yaygınlaşmasıyla akıllı test sayfası üretimi çalışmaları da araştırmaların odak noktalarından biri haline gelmektedir. Bilgisayar tabanlı akıllı test sayfası oluşturma işlemi hem sınav uygulayıcılarının iş yükünü azaltmakta hem de öğrencilerin öğrenme etkisini etkili bir şekilde kontrol etmektedir. Test sayfası üretimi çeşitli kısıtlamalar arasında dengeli olarak soruların seçimini gerektiren çok amaçlı bir optimizasyon problemi olarak ele alınabilir.

Test sayfası sorularında her bir soruya ait zorluk derecesi, soru puanı, bilgi puanı, seçilme sıklığı, çözüm süresi, yetenek seviyesi, vb. birçok öznitelik bulunmaktadır. *n* adet soru bulunan bir soru bankasında her bir sorunun *m* adet özniteliği, diğer bir ifadeyle *m* adet kısıtlama olduğu düşünülürse hedef matris *S* aşağıdaki gibi ifade edilebilir [7].

$$S = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} \tag{1}$$

Matriste her satır bir test sorusunu ve sorulara ait öznitelikleri temsil etmektedir. Her bir soruya ait özniteliklere bağlı olarak belirlenen kısıtlara uyacak şekilde test sorularının hazırlanması bir optimizasyon sorununu ortaya çıkarmaktadır. Çalışmada, test sayfası üretimi için soruların en önemli özniteliklerden olan zorluk seviyesi, soru puanı, bilgi puanı, seçilme sıklığı ve cevaplama süresi değerleri ve bu değerlerin test içindeki ağırlık katsayıları dikkate alınmıştır. Her bir özniteliğe ve test için istenen ölçütlere bağlı olarak hesaplanan hata değerleri aşağıda belirtilmiştir.

$$e_1 = \left| Dd - \frac{\sum_{i=1}^{n} a_{i1} a_{i2}}{\sum_{i=1}^{n} a_{i2}} \right| \tag{2}$$

$$e_2 = \left| Dk - \frac{\left(\sum_{i=1}^{n} a_{i3}\right)}{n} \right| \tag{3}$$

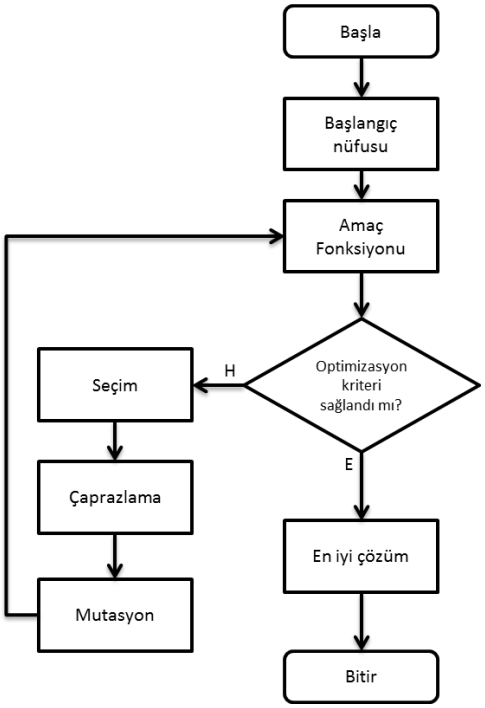
$$e_3 = \left| \frac{\left(\sum_{i=1}^{n} a_{i4}\right)}{n} \right| \tag{4}$$

$$e_4 = \left| \frac{Dk - \left(\sum_{i=1}^{n} a_{i5}\right)}{n} \right| \tag{5}$$

Burada; e_1 sorunun zorluk seviyesi için hesaplanan hata değerini (2), e_2 sorunun bilgi puanı için hesaplanan hata değerini (3), e_3 soruların seçilme sıklığının en az olması için hesaplanan hata değerini (4), e_4 ise soruların toplam süresi ile istenen süre arasındaki hata değerini ifade etmektedir (5). *Dd* istenilen zorluk derecesi faktörü, *Dk* istenilen bilgi puanı faktörü ve *Ds* ise istenilen toplam süre faktörüdür. Sorulara ait öznitelikler; a_{i1} sorunun zorluk derecesini, a_{i2} soru puanını, a_{i3} sorunun bilgi puanını, a_{i4} sorunun seçilme sıklığını ve a_{i5} ise sorunun cevaplama süresini ifade etmektedir. Bilgi puanı sorunun çözülmesi için ne kadar bilgi gerektirdiği, bölüme ait bilgi seviyesini belirtmektedir. Seçilme sıklığı sorunun daha önceki test sayfalarında kullanılma oranını belirtmektedir. Test sayfası oluşturma işleminde daha önce sorulmamış olan sorulara öncelik verilmesi dikkate alınması gereken ölçütlerden biridir.

4. GENETİK ALGORİTMA İLE TEST SAYFASI ÜRETİMİ (GENERATION OF TEST PAPER WITH GENETIC ALGORITHM)

GA, biyolojik evrim kanununa göre en uygun genetik yapının hayatta kalmasına dayanan rastgele bir arama yöntemidir [8]. Karmaşık, çok kısıtlı veya kesin çözümü olmayan problemlerin çözümünde bir nüfus içerisinde sürekli iyileşen bireylerin aranması temeline göre çalışmaktadır. Tüm çalışma alanını aynı anda paralel olarak arayarak çözümlere daha hızlı bir şekilde ulaşabilme kabiliyetine sahiptir [9]. Şekil 1 GA'nın akış diyagramını göstermektedir.

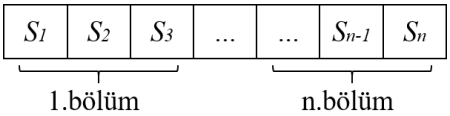


Şekil 1. *Standart GA akış diyagramı*

GA'da çözümü aranan problem için olası tüm çözümler birey (kromozom) olarak adlandırılır ve her bir birey çözümü oluşturan parametrelerden oluşmaktadır. Bu parametreler gen olarak adlandırılır. Bireylerden oluşan çözüm kümesi GA'da nüfusu oluşturmaktadır. GA'da genellikle başlangıç nüfus rastgele olarak üretilmektedir. Başlangıç nüfusunun üretilmesinden sonra her bir çözümün amaç fonksiyon değeri hesaplanmaktadır. Amaç fonksiyon değeri olası çözümün istenen çözümü ne kadar karşıladığının bir ölçüsüdür. Amaç fonksiyon değeri problemde belirlenen ölçütü sağlıyorsa algoritma sonlandırılır, aksi halde seçim, çaprazlama ve mutasyon işlemleri uygulanarak optimizasyon ölçütü sağlanana veya önceden belirlenmiş olan bir iterasyon sayısına ulaşıncaya kadar algoritma devam ettirilir. Şekil 1'de gösterilen akış diyagramında belirtilen GA'nın çalışma adımları aşağıda açıklanmaktadır.

4.1. Başlangıç Nüfusunun Oluşturulması (Initialization of Population)

GA'da başlangıç nüfusu genellikle rastgele olarak üretilmektedir. Çalışmada, her bir birey bir test sayfasını ve bireylerdeki genler de test sayfasındaki soruları temsil etmektedir. Her bölüm için soru numaraları sabit olduğundan nüfus oluşturulurken tamamen rastgele bir nüfusun oluşturulması çözüm için gereken zamanı arttırmaktadır. Bu bakımdan başlangıç nüfusu oluşturulurken tamamen rastgele değil, her bölümden istenen soru sayısına göre nüfus oluşturma işlemi gerçekleştirilmiştir. Nüfus oluşturma işleminde üretilen aday çözümler (bireyler) ikili kodlamanın daha fazla yer kaplamasından ve hesaplama süresinin artmasından dolayı tam sayı kodlama ile kodlanarak oluşturulmuştur. Şekil 2'de çalışmada *n* adet sorudan oluşan bir test sayfası oluşturma işlemi için kullanılan örnek bir bireyi göstermektedir.



Şekil 2. *Test sayfası için birey gösterimi*

4.2. Amaç Fonksiyon (Fitness Function)

Test sayfası oluşturma probleminde temel hedef; olması istenen bölümler içerisinden istenen sayıda soruların seçilmesi ve soruların özniteliklerine bağlı olarak istenen ölçütlerde bir test sayfasının oluşturulmasının sağlanmasıdır. Bölüm 3'de belirtilen her bir hata değeri ile kısıtları kullanıcı tarafından belirlenen ağırlıklarının çarpımlarının toplamı olan amaç fonksiyon değeri denklem (6) ile tanımlanmıştır.

$$f = \sum_{i=1}^{4} w_i e_i \quad (6)$$

Burada *f* amaç fonksiyon değerini, e_i *i.* kısıt fonksiyonunun hata değerini ve w_i ise *i*. kısıt fonksiyonunun ağırlığını ifade etmektedir. Kullanıcı tarafından belirlenen ağırlıkların toplamı w_t denklem (7)'de ifade edildiği gibi tüm ağırlıkların toplamı olarak ele alınmaktadır ve bu toplam 1'e eşit olacak şekilde belirlenmiştir. Test sayfası oluşturma işleminde soruların özniteliklerinin testteki etki oranı bu ağırlıklar ile belirlenmektedir.

$$w_t = \sum_{i=1}^{4} w_i \quad (7)$$

Amaç fonksiyon değerlerine bağlı olarak her bir birey için uygunluk (fitness) değeri hesaplanmaktadır. Nüfustaki bireyler için hesaplanan bu uygunluk değerlerinden 0'a en yakın olanı en iyi bireyi ifade etmektedir.

4.3. Seçim (Selection)

İyi bireylerin oluşması ve yeni nesillere aktarılması amacıyla iyi bireyleri belirlemek için bir seçim yapılması gerekmektedir. Seçim işlemi 3 adımdan oluşmaktadır; Birinci adım tüm bireylerin amaç fonksiyon değerlerinin hesaplanması, ikinci adım bireylere amaç fonksiyonu değerlerine göre uygunluk değerlerinin atanması, üçüncü adım ise bireylerin sahip oldukları uygunluk değerlerine göre seçilmeleri ve yeni birey üretimi için eşleştirme havuzuna atılmalarıdır [9]. Bu çalışmada seçim işlemi için GA'da yaygın olarak kullanılan rulet tekerleği yöntemi kullanılmıştır.

4.4. Çaprazlama (Crossover)

Çaprazlama işleminde iki ebeveyn bireyin özellikleri (genleri) birleştirilerek yeni bir birey oluşturulur [9]. Çaprazlama işleminin sonucunda, ebeveynlerinden iyi özellikleri alarak oluşan yeni bireyin ebeveynlerinden daha iyi özelliklere sahip olması beklenmektedir.

Çaprazlama sonucunda oluşacak yeni bireyde bölümler için istenilen soru sayılarının değişmeden kalması gerekmektedir. Bu amaçla çalışmada, çaprazlama işlemi için standart GA'daki çaprazlama işleminden farklı olarak bölüm tabanlı çaprazlama kullanılmıştır. Çaprazlama işleminde eşleştirilmiş olan ebeveyn bireyler için çaprazlama noktası belirlemek yerine bireylerdeki bölüm soruları kendi arasında gruplanmış ve bölümler için ayrı ayrı olmak kaydıyla rastgele çaprazlama noktaları belirlenmiştir. Bu durumda iki ebeveyn bireyin çaprazlama işlemi sırasında ebeveynlerin gen değişimleri sadece bölümler arasında yapılmaktadır. Oluşturulacak yeni bireyde birden fazla aynı sorudan oluşmaması için sorunun ebeveynden alınmadan önce yeni bireyde olup olmadığı kontrol edilir, soru yeni bireyde mevcut ise bu durumda diğer ebeveynden alınır. Şekil 3 çalışmada kullanılan çaprazlama işlemini göstermektedir.

	1.bölüm			2.bölüm			3.bölüm			
ebeveyn 1	**21**	**54**	**78**	**102**	**178**	**191**	**233**	**243**	**256**	**278**
ebeveyn 2	34	65	99	137	145	188	212	239	267	276
çocuk 1	**21**	65	99	**102**	**178**	188	212	239	**256**	**278**
çocuk 2	34	**54**	**78**	137	145	**191**	**233**	**243**	267	276

çaprazlama noktası

Şekil 3. *Test sayfası oluşturma için çaprazlama işlemi*

4.5. Mutasyon (Mutation)

GA'nın iteratif olarak çalışmasıyla birlikte uygulanan çaprazlama işlemleri sonucunda bireyler birbirlerine benzemeye başlayabilmektedir. Bu durum GA'nın paralel arama özelliğini kısıtlamakta ve çözümlerin yerel en iyiye takılmasına sebep olabilmektedir. Mutasyon işlemi bireylerin birbirlerine benzememesi ve nüfus içinde farklılık sağlanması amacıyla (yardımcı bir metot olarak) kullanılan bir

yöntemdir. Mutasyon işlemi genellikle bireyin genlerinin rastgele değiştirilmesi ile gerçekleştirilmektedir. Belirlenmiş olan bir mutasyon olasılık oranına göre mutasyon yapılıp yapılmayacağına karar verilir [9].

mutasyon öncesi

21 54 78 | 102 178 191 | 233 243 256 278

mutasyon sonrası

21 54 78 | 102 178 **180** | 233 243 **264** 278

Şekil 4. *Test sayfası oluşturma için mutasyon işlemi*

Şekil 4, örnek bir mutasyon işlemini göstermektedir. Standart GA'daki mutasyon işleminin uygulanması tekrarlı soruların oluşmasına veya mutasyona uğrayacak olan sorunun ait olduğu bölüm yerine başka bir bölümden soru seçilmesine sebep olabilmektedir. Çalışmada, mutasyon işleminde standart GA'dan farklı olarak mutasyona uğrayacak olan soru hangi bölümde yer alıyorsa soru havuzundan o bölüme ait rastgele bir soru seçilmektedir.

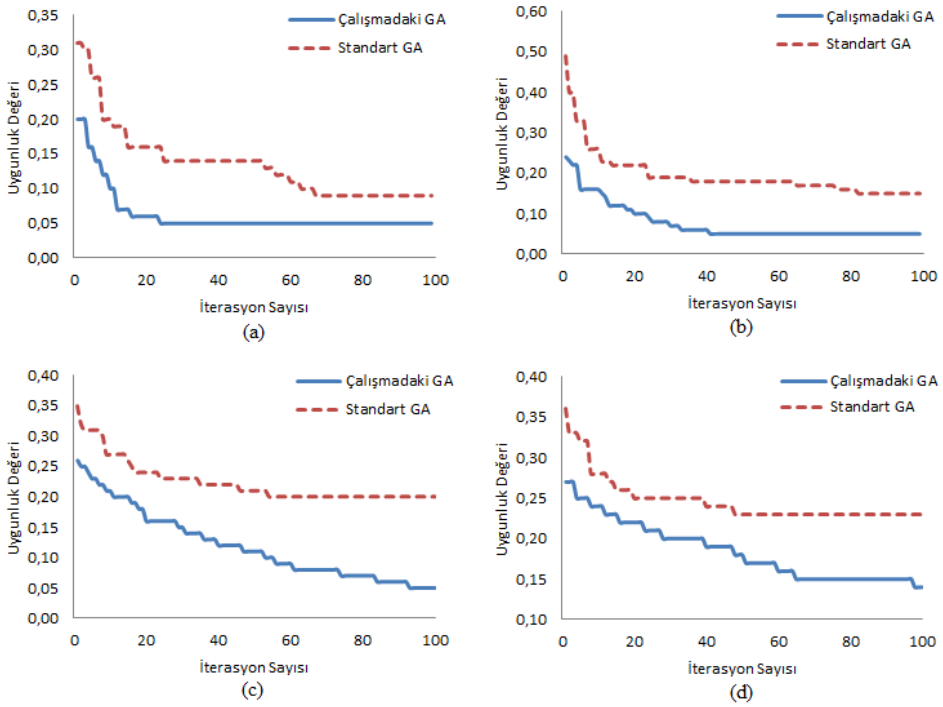
5. DENEYSEL ÇALIŞMALAR (EXPERIMENTAL STUDIES)

Çalışmada test sayfaları oluşturma işlemi için kullanılan soru bankasında toplam 2455 adet soru bulunmaktadır. 20, 50, 100 ve 200 soruluk farklı test sayfaları için standart GA ve çalışmadaki GA ile çözüm üretilmiş ve elde edilen sonuçlar karşılaştırılmıştır. Her bir test sayfası üretimi için algoritmalar 10'ar kez çalıştırılmış ve bu testlerde elde edilen uygunluk değerlerinin ortalamaları tablo 1'de gösterilmiştir. Deneysel çalışmalarda standart GA ve çalışmadaki GA için nüfus sayısı 80, iterasyon sayısı 100, çaprazlama oranı 1 ve mutasyon oranı 0,015 olarak belirlenmiştir. Soru sayısı az olan testler için nüfus sayısı daha az kullanılabilir ancak deneysel çalışmalar testteki soru sayısı arttıkça kademeli olarak nüfus sayısının artmasının daha iyi sonuçlar verdiğini ve nüfus sayısının 80'den fazla arttığında sonuçların fazla değişmediğini göstermiştir. Bu bakımdan çalışmada standardı sağlamak amacıyla tüm testler için aynı nüfus sayısı kullanılmıştır. Tablodaki veriler zorluk seviyeleri 1, 2 ve 3 için ve diğer kriterler sabit tutularak elde edilmiştir.

Tablo 1. *Standart GA ve çalışmada kullanılan GA'nın performans karşılaştırması*

Zorluk Seviyesi	Test Soru Sayısı	Nüfus Sayısı	Standart GA Uygunluk Değeri	Çalışmadaki GA Uygunluk Değeri
3	200	80	0,226	0,132
3	100	80	0,203	0,051
3	50	80	0,151	0,046
3	20	80	0,086	0,051
2	200	80	0,057	0,045
2	100	80	0,051	0,038
2	50	80	0,048	0,038
2	20	80	0,059	0,037
1	200	80	0,466	0,248
1	100	80	0,250	0,128
1	50	80	0,206	0,066
1	20	80	0,168	0,055

Soru bankası için kullanılan soru veri seti [10]'dan alınmıştır. Soru bankasında zorluk seviyesi 0 için 10, zorluk seviyesi 1 için 937, zorluk seviyesi 2 için 779, zorluk seviyesi 3 için 536, zorluk seviyesi 4 için 193 olmak üzere toplamda 2455 soru bulunmaktadır.

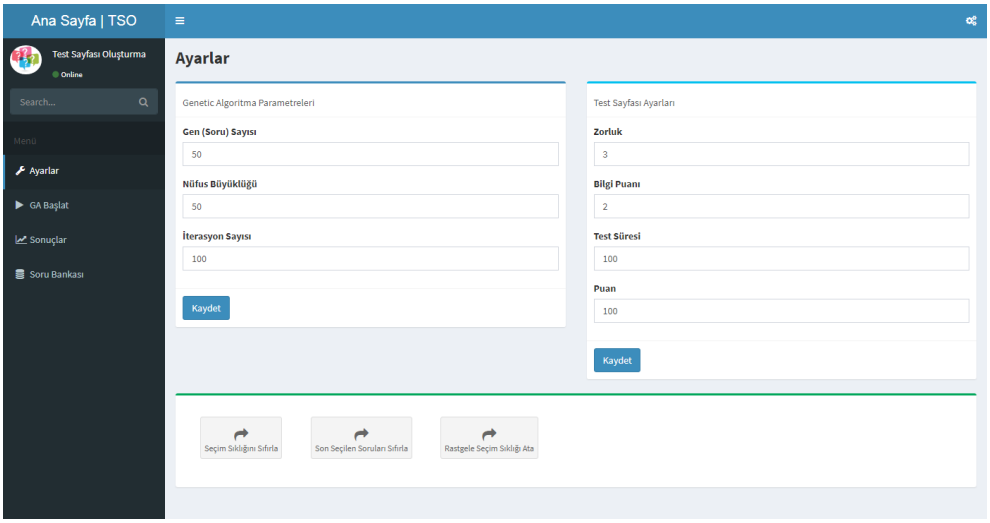


Şekil 5. *(a) 20 soruluk (b) 50 soruluk (c) 100 soruluk (d) 200 soruluk testler için Standart GA ve çalışmadaki GA'nın yakınsama grafikleri*

Standart GA ile çalışmada kullanılan GA için örnek yakınsama grafikleri şekil 5'de verilmektedir. Grafiklerden de anlaşılacağı gibi çalışmada kullanılan GA'nın standart GA'ya göre çözüme daha hızlı yakınsadığı ve daha iyi uygunluk değerlerine sahip olduğu görülmektedir.

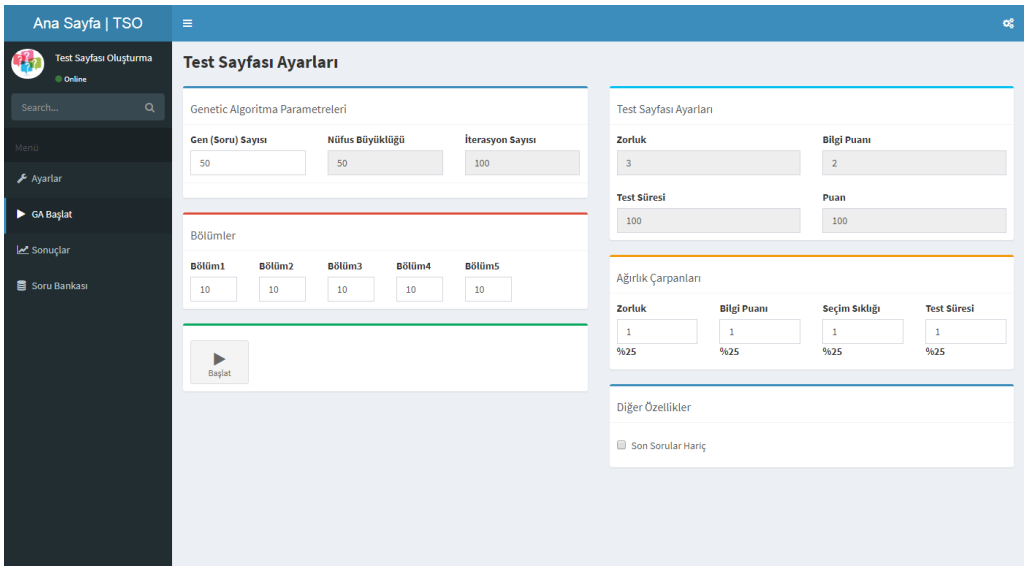
6. KULLANICI ARAYÜZÜ TASARIMI (USER INTERFACE DESIGN)

Test sayfası oluşturma için geliştirilen web tabanlı kullanıcı arayüzü şekil 6 ve şekil 7'de gösterilmektedir. Şekil 6'daki arayüzde GA için gerekli olan parametrelerin kullanıcı tarafından belirlenebildiği bir bölüm bulunmaktadır. Yine aynı arayüzde test sayfası için gerekli olan testin zorluk seviyesi, bilgi puanı, testin cevaplama süresi ve puan kısıtlarının belirlendiği bölüm bulunmaktadır.



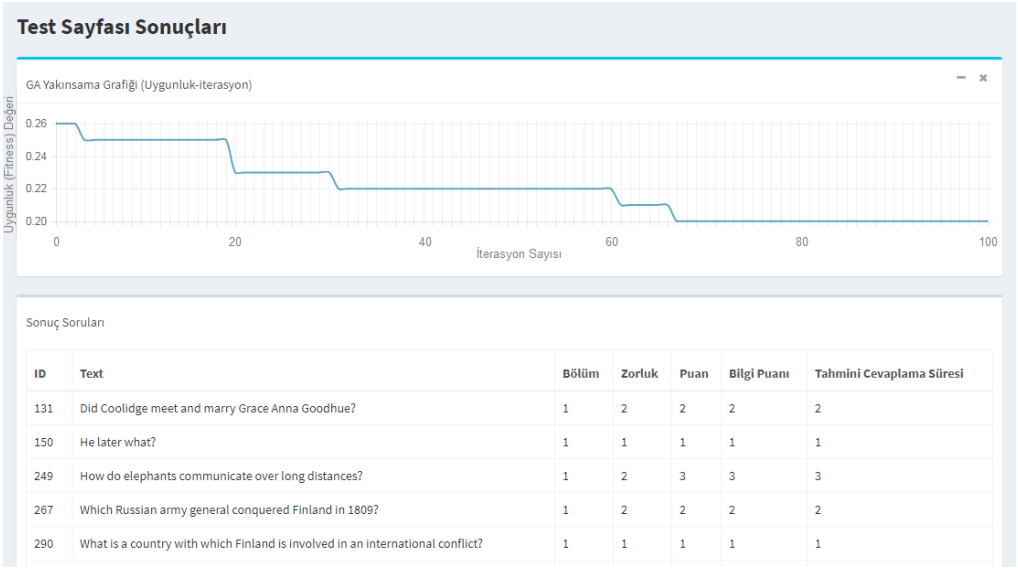
Şekil 6. *Test sayfası için arayüz-1*

Şekil 7'deki arayüzde hangi bölümden kaç adet soru olması gerektiğinin kullanıcı tarafından girilebilmesini sağlayan bir bölüm bulunmaktadır. Yine aynı arayüzde soruların özniteliklerinin testteki ağırlıklarının belirlendiği bir bölüm bulunmaktadır.



Şekil 7. *Test sayfası için arayüz-2 (User interface 2 for the test paper)*

Şekil 8'de arayüzde gerçekleştirilen test oluşturma uygulaması için sonuç ekranı görülmektedir. Bu ekranda GA'nın yakınsama grafiği ve test için seçilen soruların bilgisi kullanıcıya verilmektedir.



Şekil 8. *GA-test sayfası sonuç arayüzü (User interface of the GA-test paper result)*

7. SONUÇ (CONCLUSION)

Bu çalışmada, bir soru bankası içinden akıllı test sayfası oluşturma problemi GA kullanılarak çözülmüştür. Akıllı test sayfası oluşturma işleminde soruların zorluk seviyesi, soru puanı, bilgi puanı, seçilme sıklığı ve cevaplama süresi parametreleri dikkate alınarak çözüm üretilmiştir. Bu parametrelerin test içerisindeki ağırlıkları da dikkate alınarak test sayfaları oluşturulmuştur. Çaprazlama ve mutasyon operatörlerinde yapılan değişiklikler ile GA'nın performansının artması ve istenen kalitede test sorularının üretilmesi sağlanmıştır. Deneysel çalışmalarla standart GA ile arasındaki performans farkı ortaya konulmuştur. Aynı koşullar altında yapılan deneysel testler sonucunda, çaprazlama ve mutasyon operatörleri değiştirilen GA'nın standart GA'ya göre daha iyi sonuçlar verdiği gözlemlenmiştir. Yapılan çalışmada, kullanıcıların GA ve test sayfası için istediği ölçütleri belirleyebildiği ve algoritmayı çalıştırabildiği web tabanlı bir kullanıcı arayüzü geliştirilmiştir.

8. KAYNAKLAR (REFERENCES)

[1] Xiumin C., Dengcai W., Meining Z., Yanping Y., "Research on Intelligent Test Paper Generation Base on Improved Genetic Algorithm", The 6th International Conference on Computer Science & Education (ICCSE), IEEE, 269-272, August 3-5 2011.

[2] Jun N., "An improved genetic algorithm for Intelligent test paper generation", Intelligent Computation Technology and Automation (ICICTA), 7th International Conference on IEEE, 72-75, October 2014.

[3] Zhang K., Zhu L., "Application of Improved Genetic Algorithm in Automatic Test Paper Generation", Chinese Automation Congress (CAC), IEEE, 495-499, November 2015.

[4] Sun X., "Study on Test Databank Construction And Algorithm of Test Paper Generation System", Second International Symposium on Electronic Commerce and Security (ISECS), IEEE, 297-302, May 2009.

[5] Shan Y., "The Research and Realization of Multi-threaded Intelligent Test Paper Generation Based on Genetic Algorithm", International Conference on Computer and Information Application (ICCIA), IEEE, 461-464, 2010.

[6] Xiong L., Shi J., "Automatic Generating Test Paper System Based On Genetic Algorithm", Second International Workshop on Education Technology and Computer Science, IEEE, 2010.

[7] Wu X., Song Y., "Research on Intelligent Auto-generating Test Paper Based on Improved Genetic Algorithms", International Conference on Computational Intelligence and Software Engineering, International Conference on IEEE, December 2009.

[8] Goldberg D. E., Genetic Algorithms in Search, Optimization and Machine Learning, 1st ed., Addison-Wesley Publishing Company Inc., Boston, MA, USA, 1989.

[9] Tuncer A, Yildirim M. "Dynamic path planning of mobile robots with improved genetic algorithm", Computers & Electrical Engineering, Cilt 38, No 6, 1564-1572, 2012.

[10] İnternet: "Carnegie Mellon University, Question-Answer Dataset", http://www.cs.cmu.edu/~ark/QA-data/, Son Erişim Tarihi: 01.08.2017.