

DERS ÇİZELGELEME PROBLEMİNİN MELEZ GENETİK ALGORİTMALAR İLE PERFORMANS ANALİZİ

Mustafa GERŞİL*
Türker PALAMUTÇUOĞLU**

ÖZ

Ders çizelgelemenin oluşturulması, eğitimin aksamadan yapılabilmesi ve sonuçlarının değerlendirilebilmesi için gerekli olan en önemli hizmetlerden biridir. Ders programları belirli kısıtlar altında derslerin, uygun dersliklere ve zaman dilimine atanmasını sağlayan çizelgelerdir. Genel ders çizelgeleme problemi NP-zor (Non polinomial – zor) sınıfındadır. Bu nedenle bu problemlerin çözümünde deterministik yöntemler başarılı olamamaktadır.

Bu çalışmada, eğitimde zamanlama konusunun alt bölümü sayılabilecek ders programlarının verimliliğinin artırılması için yapay zeka yöntemlerinden biri olan genetik algoritma kullanılarak, ders programı hazırlayan ve optimize eden bir yazılım programı geliştirilmiştir. Daha etkin bir eğitim ve öğretim sağlamak için geliştirilen yazılım programı, eğitim ve öğretim verimliliği ve ders ağırlığı kriter olarak kullanılmaktadır.

Anahtar Kelimeler: Ders Çizelgeleme, Genetik Algoritmalar, Yerel Arama Teknikleri.

Jel Kodu: C40, I21

PERFORMANCE ANALYSIS OF COURSE SCHEDULING PROBLEM BY HYBRID GENETIC ALGORITHMS

ABSTRACT

Constitution of course scheduling is one of the most important services that are necessary for performing the education without interruption and assessing the results. Course programs are schedules that assign courses to appropriate classrooms and time pieces under certain constraints. General course scheduling problem is NP-hard. Therefore; the solution of these problems do not succeed by deterministic methods.

In this study, for increasing the efficiency of course programs that can be regarded as sub-section of the issue of timing in education, a software program is developed that establishes and optimizes the course program by using genetic algorithm which is one of artificial intelligence methods. The software developed in order to ensure more effective education and training use education and training productivity and the weight of the course as a criteria.

Keywords: Course Scheduling, Genetic Algorithms, Local Search Techniques.

JEL Code: C40, I21

GİRİŞ

Eğitim kurumlarında tüm hizmet işletmelerinde olduğu gibi hizmetlerin doğru bir şekilde amaçlara uygun planlanması verimlilik ve rekabet açısından kritik bir öneme sahiptir. Bu nedenle eğitim kurumlarında uzun, orta ve kısa vadede hizmet planlarının

* Yrd. Doç. Dr., Celal Bayar Üniversitesi İİBF İşletme Bölümü, mustafa.gersil@bayar.edu.tr

** Öğr. Gör. Celal Bayar Üniversitesi Kula Meslek Yüksekokulu İşletme Yönetimi Programı, btp_72@yahoo.com.tr

oluşturulması ve uygulanması zorunludur. Bu nedenle, eğitim kurumlarında, en önemli planlama faaliyetlerinden biri ders programlarının (çizelgelerinin) oluşturulmasıdır. 1960'lı yıllardan beri araştırmacıların bu probleme ilgi duyduğu ve çözümü için çeşitli yöntemler geliştirdikleri ve denedikleri görülmüştür. Bu problemin çözümünde kullanılan çeşitli yaklaşımlar grafik tabanlı teknikler, kısıt tabanlı teknikler, yerel arama teknikleri, popülasyon tabanlı teknikler, çok ölçütlü teknikler, melez ve üst sezgisel teknikler olarak sınıflandırılabilir. Son yıllardaki çalışmaların önemli ölçüde, çok ölçütlü, melez ve üst sezgisel tekniklere yöneldiği söylenebilir.

Zaman çizelgeleme problemlerinde belirli bir veya daha fazla sayıda amaç ve çeşitli katı ve yumuşak kısıtlar mevcuttur. Katı kısıtlar oluşturulacak çizelgelerin olurlu olması için mutlaka sağlanması gereken kısıtlardır. İhlal edilen herhangi bir katı kısıt çizelgenin olursuz olmasına neden olur. Yumuşak kısıtlar ise, çizelgenin olurlu olması bakımından mutlaka sağlanması gerekmeyen, ancak çizelgenin kalitesini belirleyen kısıtlardır. Yumuşak kısıtlar ne ölçüde yerine getirilebilirse elde edilen çizelgeler o ölçüde kaliteli olacaktır. Yumuşak kısıtların ihlal edilmesi hizmeti alanları da, hizmet verenleri de, işletme yönetimini de olumsuz yönde etkileyerek hizmetin algılanan kalitesini düşürebilir. Bu nedenle zaman çizelgeleme problemleri genellikle yumuşak kısıt ihlallerinin azaltılması amacı ile modellenmektedir. Ders çizelgeleme bir dizi üniversite dersinin tamamı için ortak öğrencileri olan derslerin çakışmalarını minimize ederek haftalık çizelgelerin oluşturulması problemidir (Schaerf, 1995:2). Ders çizelgeleme derslere, bölümlere veya sınıflara atanan öğrenciler ve öğretmenlerin olduğu çok yönlü bir atama problemidir (Carter ve Laporte,1998:4).

Bu çalışmada ders programlarının otomatik çizelgelemesini yapan bir bilgisayar programı geliştirilmiştir. Program "Kullanıcı Arabirimi", "Veri Tabanı" ve "Sezgisel Çözümleyici" ismini verdiğimiz üç ana bileşene sahiptir. Kullanıcı arabirimi kullanıcı ile veritabanı ve çözümleyici arasındaki bilgi alış verişini yönetmeyi sağlamaktadır. Veri tabanı, kullanıcı arabiriminden ya da sezgisel çözümleyiciden gelen bilgileri depolar ihtiyaç duyulduğunda gerekli bilgileri kullanıcı arabirimine ya da sezgisel çözümleyiciye iletir. Sezgisel çözümleyici veri tabanından aldığı bilgileri kullanarak problem için en iyi çözümü bulmaya çalışır ve elde ettiği en iyi çizelgeyi veri tabanına ileterek depolanmasını sağlar. Kullanıcı ara birimi vasıtasıyla probleme ilişkin temel verilerin ve kısıtların veri tabanına kaydedilmesi sağlanır. Daha sonra sezgisel çözümleyicinin çalıştırılması ve elde edilen çözümlerin bölüm, sınıf, öğretim elemanı ya da gün bazında raporlanabilmesini sağlar.

1. DERS ÇİZELGELEME PROBLEMİ

Bir ders çizelgesinde ders aktivitelerinin derslik ve otuurlara atanması gerekir. Ancak bu atama yapılırken katı kısıtların kesinlikle karşılanması, yumuşak kısıtların ise mümkün olduğunca karşılanması gerekir (Palamutçuoğlu, 2008: 126).

Ele alınan probleme ilişkin katı kısıtlar aşağıda sıralanmıştır:

K.1. Bir öğretim elemanının verdiği derslerin aynı oturuma denk düşmemesi gerekir.

K.2. Bir öğrenci grubunun şubelere ayrılan dersler hariç alacağı derslerin aynı oturuma atanmaması gerekir. Öğretim elemanları farklı olan iki farklı şubenin dersi aynı oturuma atanabilmelidir.

K.3. Bir dersliğe aynı oturumda birden fazla ders atanmaması gerekir.

K.4. Standart dersliklerin yanı sıra özel amaçlar için tasarlanmış derslikler de mevcuttur. Bazı dersler için standart derslikler yeterli olurken, bazı dersler için özel derslikler tercih edilmektedir. Fakültede dört farklı derslik türü mevcuttur: Standart derslikler, amfiler, projeksiyonlu derslikler, bilgisayar laboratuvarları. Her derslik türü için herhangi bir oturma atanan derslerin sayısı, bu türdeki derslik sayısını aşmamalıdır.

Probleme ilişkin yumuşak kısıtlar ise aşağıdaki gibidir:

Y.1. Oturlara atamalarda, öğretim elemanlarının tercihleri mümkün olduğunca yerine getirilmelidir. Öğretim elemanlarının dersleri mümkün olduğunca istediği gün ve saatlere atanmaya çalışılmalı, istemediği gün ve saatlere ise atanmamalıdır.

Y.2. Derslik atamalarında da benzer şekilde, öğretim elemanlarının tercihleri mümkün olduğu kadar yerine getirilmelidir.

Y.3. Dersleri alan öğrenci sayıları dersliklerin kapasitesini mümkünse aşmamalı, mümkün değilse en az ölçüde aşmalıdır.

Y.4. Şubelere ayrılan dersler aynı öğretim elemanı tarafından veriliyorsa, aksi belirtilmemişse mümkün olduğu ölçüde aynı güne ardışık olarak atanmaya çalışılmalıdır.

Kısıtlamalardan anlaşıldığı gibi katı kısıtlar yerine getirilmediği takdirde geçerli bir çizelge oluşturulamamaktadır. Çünkü söz konusu katı kısıtlar fiziki açıdan kaynakların kullanımına ilişkin kısıtlardır. Yumuşak kısıtlar ise daha ziyade oluşturulan çizelgenin kalitesi ile ilgilidir. Örneğin, öğretim elemanları istemedikleri oturlara veya dersliklere atanırlarsa çizelgeden hoşnut olmazlar. Benzer şekilde dersi alan öğrenci sayıları atandıkları dersliğin kapasitesinin çok üstünde olursa, dersin verimli bir şekilde yapılabilmesi mümkün olmaz ve hem öğretim elemanı, hem de öğrenciler bu durumdan rahatsız olurlar. Şubelere ayrılan derslerin aynı gün olması öğretim elemanının ders dışı zamanını arttırmakta ve akademik faaliyetleri için daha fazla zaman ayırabilmesini sağlamaktadır.

1.1. Ders Çizelgeleme Probleminin Matematiksel Modeli

Ders çizelgeleme çok sayıda katı ve yumuşak kısıtı sağlamaya çalışarak ders faaliyetlerinin derslik ve oturlara atanması problemi olduğundan genellikle optimizasyon problemi şeklinde modellenir. Bu tür modellemede en iyileştirilecek bir amaç fonksiyonu kullanılır ve kısıtlamalarda eşitlik veya eşitsizlik şeklinde gösterilir.

D:	{D ₁ , D ₂ ,, D _i }	Dersler
DM:	{DM ₁ , DM ₂ ,, DM _i }	Derslerin öğrenci mevcutları
S:	{S ₁ , S ₂ ,, S _s }	Sınıflar (aynı müfredatı paylaşan öğrenci grupları)
SD:	{SD ₁ , SD ₂ ,, SD _s }	Sınıfların bütün dersleri
SDa:	{SD _{s1} , SD _{s2} ,, SD _{sa} }	Sınıfların şubelere ayrılmamış dersleri
SDB:	{SDB ₁ , SDB ₂ ,, SDB _s }	Sınıfların şubelere ayrılmış dersleri
O:	{O ₁ , O ₂ ,, O _o }	Öğretim elemanları
TD:	{TD ₁ , TD ₂ ,, TD _o }	İlgili öğretim elemanının dersleri
C:	{C ₁ , C ₂ ,, C _a }	Derslikler
CK:	{CK ₁ , CK ₂ ,, CK _a }	Derslik kapasiteleri
DT:	{DT ₁ , DT ₂ ,, DT _r }	Dersliklerin farklı tipleri
OC:	{OC ₁ , OC ₂ ,, OC _r }	r Tipi derslikler
ODC:	{ODC ₁ , ODC ₂ ,, ODC _r }	r Tipi derslik gerektiren dersler
Z:	{Z ₁ , Z ₂ ,, Z _t }	Oturlar
m _{co} :		O tipi derslik sayısı
m _c :		Derslik sayısı.

P_{ita} : D_i dersinin C_a dersliğine ve Z_t periyoduna atanması için tercih ağırlığı (Öğretim elemanlarının tercihlerine göre belirlenmektedir.)

$$X_{ita} = \begin{cases} 1 & \text{Eğer } D_i \text{ dersi } Z_t \text{ periyoduna ve } C_a \text{ dersliğine atanırsa} \\ 0 & \text{Aksi Durumda} \end{cases}$$

1.2. Problemin Formülasyonu

Arama problemlerinde katı kısıtları tatmin eden bir çözümün bulunması yeterlidir. Bu çözüme olurlu çözüm adı verilmektedir. Optimizasyon probleminde ise maksimum ya da minimum olması istenilen bir amaç denklem bulunmaktadır. Ders çizelgeleme problemlerinde sadece olurlu çözüm bulmak yeterli değildir. Bu nedenle problem bir optimizasyon problemi olarak formüle edilmiştir.

Amaç Denklem:

$$\text{Mak } Z = w_1 \sum_{i \in D} \sum_{r \in Z} \sum_{a \in C} P_{ita} \cdot X_{ita} + w_2 \sum_{i \in D} \sum_{r \in Z} \sum_{a \in C} (CK_a - DM_i) \cdot X_{ita} + w_3 \cdot \sum_{i \in SDB_s} \left(\sum_{\tau=t}^{t+1} X_{ita} \right) - 2$$

K.1. Öğretim elemanlarına ait derslerin çakışmaması gereklidir:

$$\sum_{t \in Z} \sum_{a \in C} X_{ita} \leq 1 \quad (\forall o \in O, \forall i \in TD_o)$$

K.2. Sınıfların derslerinin çakışmaması gerekmektedir. Eğer derslerin şubelere ayrılmayacağı varsayılırsa kısıt aşağıdaki gibi formüle edilir:

$$\sum_{a \in C} \sum_{t \in Z} X_{ita} \leq 1 \quad (\forall s \in S, \forall i \in SD_s)$$

Ancak ele alınan ders çizelgeleme probleminde derslerin bazılarının şubelere ayrıldığı, bazılarının ise şubelere ayrılmadığı görülmüştür. Bir dersin sadece iki şubeye ayrılacağı varsayılmaktadır. Bu durumda sınıfların ders çakışmasını önlemek için daha farklı bir kısıt yapısı kullanmak gerekir. Bu kısıtta eğer şubelere ayrılmamış bir derse herhangi bir oturum için atama yapılmışsa aynı oturuma başka ders atanmaması gerekir. Aksi durumda şubelere ayrılan iki farklı dersin farklı şubeleri öğretim elemanları da farklı ise aynı oturuma atanmasına izin verilmelidir. Bu durum için iki farklı kısıt olmalıdır:

Bir oturuma sadece iki ayrı şubenin dersi atanabilir:

$$\sum_{i \in SDB_s} \sum_{a \in C} X_{ita} \leq 2 \quad (\forall s \in S, \forall t \in Z)$$

Bir oturuma şubelere ayrılmamış bir ders atanırsa başka ders atanmamalıdır:

$$\sum_{a \in C} \sum_{i \in SDA_s} X_{ita} + \text{Round} \left[\frac{\sum_{a \in C} \sum_{i \in SD_s} X_{ita}}{2} \right] \leq 1 \quad (\forall s \in S, \forall t \in Z)$$

K.3. Her-hangi bir dersliğe herhangi bir oturumda sadece bir ders atanmalıdır:

$$\sum_{i \in D} \sum_{t \in Z} X_{ita} \leq 1 \quad (\forall a \in C)$$

K.4. Her derslik tipi için herhangi bir oturuma atanan ders sayısı o tipteki derslik sayısından fazla olmamalıdır:

$$\sum_{a \in OC_r} \sum_{i \in ODC_r} X_{ita} \leq mco_r \quad (\forall r \in DT, \forall t \in Z)$$

Model öncelikle bir arama problemi olarak formüle edilmiştir. Daha sonra ise yumuşak kısıtları en iyi ölçüde tatmin edecek bir amaç fonksiyon ile optimizasyon problemi olarak formüle edilmiştir.

2. GENETİK ALGORİTMA

Standart genetik algoritma, gerçek problemlerin çözümünde pek fazla performans gösterememektir. Bu algoritmadaki temel operatörlere diğer operatörler, yani birimler ilave edilerek temel genetik algoritmanın performansı geliştirilebilir. “Grefenstette (1986) tarafından tanımlanan böyle bir genetik algoritma çoğu uygulayıcı tarafından oldukça yaygın kullanılmaktadır” (Grefenstette, 1986: 122-128).

Genetik algoritmanın temel adımları (Karaboğa, 2004: 88):

1. *Muhtemel çözümlerden başlangıç popülasyonunu oluştur.*
2. *Popülasyondaki her çözümün uygunluk değerini hesapla.*
3. *Durdurma kriteri sağlanıyorsa araştırmayı durdur. Yoksa aşağıdaki adımları gerçekleştir.*

- *Uygunluk ölçekleme işlemini gerçekleştir (fitness scaling).*
- *Tabii seçim işlemini uygula (uygunluk değerleri daha yüksek olan çözümler yeni popülasyonda daha fazla temsilciye sahip olacaktır)*
- *Rasgele seçim işlemini uygula (popülasyonun çoğunluğu tabii seçim, az bir kısmı da rasgele seçim işlemine tabi tutulur).*
- *Önceki popülasyonda var olan en iyi çözümü muhafaza et (tabii olmayan bir işlem).*
- ~~Çaprazlama işlemini uygula.~~
- Mutasyon işlemini uygula.

3. ARAMA ALGORİTMALARI

Standart genetik algoritmanın bölgesel yakınsama hızının çok iyi olmaması önemli dezavantajlarından birisidir. Bu eksikliği kapatmak amacıyla standart genetik algoritmayla klasik türeve dayalı algoritmaları birleştirmişlerdir. Bu tip genetik algoritmalara melez (hybrid) genetik algoritmalar ismi verilir (Karaboğa, 2004: 89).

3.1. Açgöz Rasgele Adaptif Arama Prosedürü

Aç göz rastgele adaptif arama prosedürü (ARAAP), her iterasyonu yapım ve yerel arama adımlarından oluşan çok adımlı ve iteratif bir süreçtir. Yapım aşaması olurlu bir çözüm oluşturur. Yerel arama esnasında bir yerel minimum/maksimum bulununcaya kadar mevcut çözümün komşuları araştırılır. Tüm çözümlerin en iyisi sonuç olarak saklanır. Yapım aşamasının her iterasyonunda, olurluluğu tahrir etmeden kısmi çözüme dahil edilebilecek tüm öğelerin aday öğeler kümesine yerleştirilmesi sağlanır. Çözüme dahil edilecek yeni öğenin seçimine tüm aday öğelerin açgözlü bir değerlendirme fonksiyonuna göre değerlendirilmesi sonucunda karar verilir (Nehi ve Gelareh, 2007: 2307).

Bu açgözlü fonksiyon oluşturulan çözüme bu öğelerin dahil edilmesinden dolayı maliyet fonksiyonundaki marjinal gelişimi gösterir. Öğelerin bu fonksiyon ile değerlendirilmesi en iyi öğeler ile oluşturulan bir sınırlandırılmış aday listesi oluşturulmasına yol açar. İlgili öğelerin mevcut kısmi çözüme dahil edilmesi en küçük marjinal maliyeti sağlar (bu sezgiselin açgözlü durumudur). Kısmi çözüme dahil edilecek öğe bu sınırlandırılmış aday listesinden rastgele seçilir (bu sezgiselin olasılıklı-rastgele durumudur). Kısmi çözüme dahil edilecek öğe seçilince aday listesi güncellenir ve marjinal maliyet yeniden değerlendirilir (bu sezgiselin adaptif durumudur). “Bu strateji Hart ve Shogan’ın yarı-açgözlü sezgiseline ve yerel aramasız rasgele açgözlü yapıma benzer. Bu rasgele açgözlü seçim algoritmasının sözde kodu aşağıda gösterilmektedir” (Nehi ve Gelareh, 2007: 2308).

Function Açgöz_Rasgele_Oluşturma

1. Çözüm $\leftarrow \emptyset$
2. *Aday elemanların marjinal maliyetlerini değerlendir*
3. *While çözüm tamamlanmadıysa Do*
4. *Sınırlandırılmış aday listesini oluştur*
5. *Listeden rasgele bir öğe seç (x)*
6. Çözüm \leftarrow Çözüm $\cup \{x\}$ (Adayı kısmî çözüme dahil et)
7. *End*
8. *Çözümü Döndür*
9. *End Açgöz_Rasgele_Oluşturma*

Açgözlü rasgele yapım ile oluşturulan çözümler optimal değildir. Yerel arama adımı genellikle oluşturulan çözümü iyileştirir. Yerel arama algoritması her iterasyonda mevcut çözüm ile en iyi komşu çözümün değiştirildiği iteratif bir yapıda çalışır. Yerel aramanın etkinliği, komşuluk yapısı, komşuluk arama tekniği, komşuların hızlı maliyet değerlendirilmesi ve başlangıç çözümünün kalitesi gibi bazı etkenlere bağlıdır. Yapım aşaması, yerel arama için yüksek kalitede başlangıç çözüm oluşturmada önemli bir rol oynar. Komşuluk arama *en iyi gelişim* ve *ilk gelişim* stratejilerine göre yapılabilir (Nehi ve Gelareh, 2007: 2309).

3.2. Tepe Tırmanma Algoritması

Tepe tırmanma (TT) metodu bir iteratif iyileştirme (yerel arama) yöntemidir. TT, iniş ya da adım adım iniş stratejisi olarak da adlandırılmaktadır. Bu metodun temelinde, tanımlanan bazı kurallara göre bir çözümden bir diğer komşu çözüme ulaşma vardır. TT metodunun etkinliğinde iyi bir komşuluk yapısı seçiminin önemi büyüktür. İyi mutlak açıdan en iyi olmak zorunda değildir. TT’nin zayıf yanı yerel ve genel en iyi arasındaki ayrımı yapamaması sonucu yerel optimumdan kaçamamasıdır. Yerel optimumdan genel optimuma geçebilmek için modern sezgiseller geliştirilmiştir (Yiğit ve Türkbey, 2003: 49).

Algoritmanın sözde kodu aşağıdaki gibidir:

- X: Çözüm Uzayı,
- x: Aday çözüm,
- x*: Cari en iyi çözüm,
- S(x): Komşuluk üretme hareketleri,

$s(x)$: Bir komşuluk üretme hareketi,
 $s_k(x)$: k.iterasyonda elde edilen en iyi komşu üretme hareketi,
 k : İterasyon sayacı,
 $c(x)$: x çözümünün amaç fonksiyon değeri,
 $c(x^*)$: Cari çözümün amaç fonksiyon değeri.

1. $x \in X$ olacak şekilde bir başlangıç çözümü seç ve en iyi çözüm olarak kabul et ($x_* = x$). İterasyon sayacını sıfırla ($k = 0$).
2. Aday hareketlerden en iyisini seç $S_k(x) = \text{Optimum} (s(x) \in S(x))$.
İterasyon sayacını bir arttır ($k = k + 1$).
3. $x = s_k(x)$ olarak ata. Eğer x' in amaç fonksiyon değeri $c(x)$, x^* in amaç fonksiyon değerinden $c(x^*)$ daha iyiyse (Minimizasyon için $c(x) < c(x^*)$ ise) optimal çözümü güncelle ($x^* = x$)
4. Eğer maksimum iterasyon sayısına ulaşıldı ise veya çözüm daha fazla iyileştirilemiyorsa araştırmayı durdur.

3.3. Tabu Arama Algoritması

Tabu araştırma algoritması (TA), Glover tarafından optimizasyon problemlerinin çözümü için geliştirilmiş iteratif bir araştırma algoritmasıdır (Glover, 1989, 1990, 1993). "TA algoritması bölgesel-sezgisel araştırma performansının artırılması amacıyla, zeki problem çözenin genel kurallarını bu klasik araştırmaya kılavuzluk edecek şekilde kullanmaktadır. Başka bir deyişle TA algoritması, bölgesel en iyi çözümün daha ilerisinde bulunan çözümleri araştırabilmesi için bölgesel-sezgisel araştırmaya kılavuzluk yapmaktadır (Chelouah ve Siarry, 2000). TA'nın bölgesel optimalite aşmak amacıyla kullandığı temel prensip, değerlendirme fonksiyonu denilen bir fonksiyon tarafından her iterasyonda en yüksek değerlendirme değerine sahip hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesine dayanmaktadır" (Karaboğa, 2004: 49-50).

Tabu araştırmanın altında yatan temel fikir, araştırma işlemi kontrol etmek amacıyla esnek bir hafıza yapısının kullanılmasıdır. TA'nın temel işlem basamakları aşağıda verilmektedir (Glover, 1989: 192; Glover, 1990: 4):

X : Çözüm uzayı
 x : Çözüm uzayındaki herhangi bir çözüm
 x^* : O ana kadar bulunan en iyi çözüm
 $S(x)$: x 'in yeni komşularını üretmede kullanılan hareketler kümesi
 s_k : k.iterasyonda seçilen hareket
 $s_k(x)$: s_k 'nin uygulanmasıyla elde edilen komşu çözüm
 T : Tabu listesi

1. $x \in X$ olacak şekilde şekilde bir başlangıç çözümü seç ve en iyi çözüm olarak kabul et ($x^* = x$). İterasyon sayacını sıfırla ($k = 0$) ve tabu listesini boşalt (T).
2. Tabu listesinde olmayan hareketlerin kümesi ($S(x) - T$) boş ise 4. adıma git.
Aksi durumda, iterasyon sayacını arttır ve tabu listesinde olmayan hareketlerin en

iyisini seç $S_k(x) = \text{Optimum}(s(x) - T)$.

3. $x = S_k(x)$ olarak ata. Eğer x 'in amaç fonksiyon değeri $c(x)$, x^* 'in amaç fonksiyon değerinden $c(x^*)$ daha iyiyse (minimizasyon için $c(x) < c(x^*)$ ise) $x^* = x$ olacak şekilde optimal çözümü güncelle.

4. Eğer maksimum iterasyon sayısına ulaşıldıysa, optimal çözüm bulunduysa ya da tabu olmayan hareketler kümesi boş ise araştırmayı durdur. Aksi durumda tabu listesini güncelle ve 2. adıma geri dön.

En iyi kabul edilebilir hareket, mevcut çözümün komşuluğunda bulunan amaç değeri ve tabu sınırlamaları açısından en yüksek değerlendirme değerine sahip olan harekettir. Değerlendirme fonksiyonu, amaç fonksiyonunda en az kötüleşme veya çok iyileşme sağlayan hareketi seçer. Tabu listesinde kabul edilen hareketlerin karakteristikleri depolanmakta ve bu karakteristikler sonraki iterasyonlarda tabu olacak (kaçınılacak) belirli hareketleri sınıflandırmak için kullanılmaktadır. Kötüleşme sağlayan hareketlerin kabul edilebilir olması nedeniyle daha önceden denenmiş olan çözümlere tekrar dönmek mümkün olabilmekte ve böylece bir döngü oluşabilmektedir. Tabu listesini kullanmanın amacı böyle bir döngünün oluşmasını önlemektir. Bu yüzden, görevi sadece tabu listesini kontrol etmek ve güncellemek olan bir yasaklama stratejisini kullanarak araştırmayı sınırlandırmak zorunludur (Karaboğa, 2004; 51-52).

3.4. Tavlama Benzetimi

Tavlama benzetimi (TB) algoritması orijinini doğal ısı işleminden almaktadır. Bu algoritmanın optimizasyon problemlerine uygulaması ile ilgili çalışmalar, 1983 yılında yapılan bir araştırma ile başladı. Algoritma metallerin ısı işlemi ile bir optimizasyon problemine çözüm araştırma olayları arasındaki benzerlikten ilham alınarak ortaya konulmuştur (Kirkpatrick vd., 1983: 671-680).

Güvenilir sezgisel bir araştırma algoritması başlangıç noktasına bağımlılığı az olan algoritmadır. Amaç, sonunda bir minimum bulunmasını sağlamaktır. Bu yüzden yukarı doğru olan hareketlerin kabulü belirli bir kontrol stratejisi ile yapılmalıdır. Tavlama benzetimi algoritmasında bu şekildeki hareketlerin kabul edilme sıklığı araştırma ilerledikçe değişen olasılık tabanlı fonksiyon ile kontrol edilir. Bu kontrol işlemini tanımlamada Kirkpatrick vd.'ne ilham kaynağı olan fikir, Metropolis vd.'nin istatistiksel termodinamik ile ilgili yapmış olduğu bir çalışmadan çıkmıştır (Metropolis ve diğ., 1953: 1088). Termodinamik kanunları, T sıcaklığında enerjide δE genlikli bir artışın olma olasılığını

$$p(\delta E) = \exp\left(-\frac{\delta E}{kT}\right)$$

ifadesi ile tanımlanmaktadır; burada k Boltzman sabiti ($1,38 \times 10^{-23}$ j/k) olarak bilinen bir fiziksel sabittir. Sabit T sıcaklık değeri için bir kristalin termal dengeye erişiminin benzetimini yapmak amacıyla Monte-Carlo metodunu geliştirmişlerdir (Metropolis ve diğ., 1953: 1088). Bu benzetim mevcut durumda bir pertürbasyon (bozulma) oluşturmakta ve bu değişimden dolayı ortaya çıkan enerji değişimini hesaplamaktadır. Şayet enerji azalmışsa sistem bu yeni duruma kaymaktadır. Enerji artmışsa yeni durum yukarıda verilen olasılığa göre kabul edilmektedir (Metropolis vd, 1953: 1088).

3.4.1 Tavlama Benzetimi Algoritması

Tavlama benzetimi algoritmasının temel adımları aşağıda verilmiştir (Karaboğa, 2004:37):

1. *Sezgisel olarak veya rasgele bir başlangıç çözümü x üret. Sıcaklık T için bir başlangıç değeri T_s belirle ve durdurma kriteri gibi diğer parametreleri belirle.*
2. *Komsu bir çözüm $x^* \in S(x)$ üret ve bu üretilen çözümle x çözümünün amaç değerleri arasındaki farkı $\Delta = c(x^*) - c(x)$ hesapla.*
3. *Eğer*
 - i. *x^*, x' den daha iyi (yani minimizasyon için $\Delta < 0$, maksimizasyon için $\Delta > 0$) ise veya*
 - ii. *x^*, x' den daha kötü ama mevcut sıcaklık T 'de $e^{-\Delta/T} > \theta$ ile (burada $\theta, 0 < \theta < 1$ olacak şekilde rastgele üretilmiş bir sayıdır) kabul edilmiş ise*
 O zaman x çözümünü x^ çözümü ile yer değiştir. Yoksa, x^* 'ı mevcut çözüm olarak muhafaza et.*
4. *Belirli bir kurallar setine göre T sıcaklığını değiştir. Bu,*
 - i. *Kullanılan soğutma tarifesine göre,*
 - ii. *Üçüncü adımda üretilen çözümlerde herhangi bir gelişme olup olmadığına göre,*
 - iii. *x çözümüne ait komşuluğun tamamen araştırılıp araştırılmadığına göre veya başka kriterlere göre belirlenmiş kurallar olabilir.*
5. *Durdurma kriteri sağlanıyorsa araştırmayı durdur, yoksa 2. adıma geri git.*

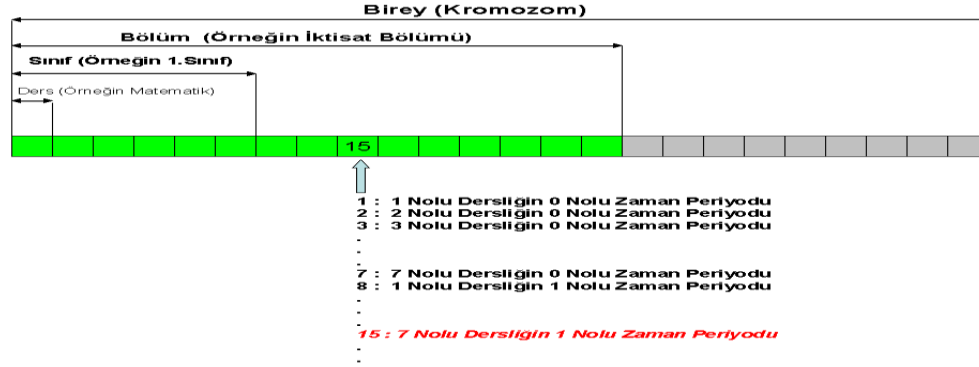
4. UYGULAMA

Bu çalışmada, Celal Bayar Üniversitesi İktisadi ve İdari Bilimler Fakültesi'nde gerekli izinler alınarak 2007-2008 eğitim-öğretim yılı güz ve bahar yarıyılarında ders programları için kullanılan veriler (dersler ve öğrenci mevcutları, derslerin şubelere ayrılıp ayrılmayacağı, özel derslik gerektirip gerektirmediği, dersleri veren öğretim elemanları, öğretim elemanlarının oturum ve derslik tercihleri, dersliklerin sayısı ve kapasiteleri, probleme ilişkin kısıtlamalar) öğrenci işleri şefliğinden alınmıştır.

4.1. Kodlama Yapısı

Geliştirilen genetik algorithmada permütasyon tipi kodlama kullanılmıştır. Bireylerin her geni bir dersi ve bu derse ilişkin diğer bilgileri (dersi veren öğretim elemanını, dersin şubelere ayrılması durumunda şubesini, öğrenci mevcudunu) temsil etmektedir. Bu genlerin bir araya gelmesi ile sınıflar ve sınıfların bir araya gelmesi ile de bölümler temsil edilmektedir. Bölümlerde bir araya gelerek bireyi oluşturmaktadır. Genlerin aldıkları değerler ise temsil ettikleri dersin atandığı oturumu ve dersliği temsil etmektedir. Başka bir deyişle genler "1" ile "oturum sayısı x derslik sayısı" arasında bir değer alabilmektedir. Bu değerlerin her biri bir dersliği ve oturumu ifade eder. Böylece

derslerin hangi oturuma ve hangi dersliğe atanacağı belirlenebilmektedir. Kullanılan kodlama yapısı Şekil 1’de gösterilmiştir.



Şekil 1: Ders Çizelgeleme Problemi İçin Geliştirilen Genetik Algoritmada Kullanılan Kodlama Yapısı

Her gen “1” ile “oturum sayısı x derslik sayısı” arasında değer olarak kodlandığından, uygunluk değerlendirilmesinde bu kodun çözülmesi gerekecektir. Kod çözümüyle dersin atandığı gün, saat ve derslik belirlenerek değerlendirme ona göre yapılacaktır. Kodun çözümlenmesi Tablo 1’deki formüllere göre yapılmaktadır:

Gün	=	(Genetik Bilgi-1) \	(derslik sayısı x bir gündeki oturum sayısı)
Oturum	=	(Genetik Bilgi-1) \	derslik sayısı
Derslik	=	(Genetik Bilgi-1) MOD	derslik sayısı

Tablo 1: Kod Çözümlemesinde Kullanılan Formüller.

4.2. Uygunluk Değerlendirme

Uygunluk değerlendirmede ödül-ceza sistemi kullanılmaktadır. Kullanıcı ara birimi aracılığıyla girilen öğretim elemanlarının oturum ve derslik tercihleri kullanılarak bir ödül puan tablosu oluşturulmaktadır. Tablo oluşturulurken tercih belirten öğretim elemanlarının yönetici konumunda olması, unvanı ve kıdemi göz önünde tutularak ödül puanı belirlenmektedir. Tablo 2’de bu puanlamada kullanılan ağırlıklar gösterilmiştir.

Unvan	Ödül Puan Ağırlığı	Kıdem	Ödül Puan Ağırlığı	Yönetim Görevi	Ödül Puan Ağırlığı
Prof. Dr.	6	20 Yıl ve Üzeri	5	Rektör	10
Doç. Dr.	5	15-19 Yıl	4	Rektör Yrd.	9
Yrd. Doç. Dr.	4	10-14 Yıl	3	Dekan	8
Öğr. Gör. Dr.	3	5-9 Yıl	2	Dekan Yrd.	7
Öğr. Gör.	2	4 Yıl ve Altı	1	Enstitü Müd.	6
Okutman	1			Enstitü Müd. Yrd.	5
				Diğer Yönetim Görevleri	4

Tablo 2: Ödül Puantaj Tablosunun Hesaplanmasında Kullanılan Ağırlıklar.

Öğretim elemanları oturum ve derslik tercihlerini 4 farklı şekilde belirleyebilirler:

- 1.tip tercih kısıt türü seçildiğinde, belirtilen oturum veya -dersliklerden herhangi birisine atama yapılması,
- 2.tip tercih kısıt türü seçildiğinde, sadece belirtilen oturum veya dersliğe atama yapılması,
- 3.tip tercih kısıt türü seçildiğinde, belirtilen oturumlara veya dersliklere atama yapılmaması,
- 4.tip tercih kısıtı seçildiğinde ise, belirtilen oturma veya dersliğe kesinlikle atama yapılmaması istenir.

Ödül puanı hesaplamada bu kısıt türlerine göre bir puan kullanılmaktadır. Kısıt türlerine verilen puanlar Tablo 3’de gösterilmiştir.

Kısıt Türü	Ödül Puanı
Ders bu zaman periyotlarına atanabilir.	5
Ders kesinlikle bu zaman periyoduna atansın.	100
Ders bu zaman periyotlarına atanmasın.	-5
Ders kesinlikle bu zaman periyoduna atanmasın.	-100
Ders bu dersliklere atanabilir	5
Ders kesinlikle bu dersliğe atansın	100
Ders bu dersliklere atanmasın	-5
Ders kesinlikle bu dersliğe atanmasın	-100

Tablo 3: Öğretim Elemanlarının Tercih Kısıt Türleri ve Ödül Puanları.

Ödül puantaj tablosu hesaplanırken, öğretim elemanının seçtiği kısıt türüne göre ödül puanı belirlenir ve öğretim elemanının unvan, kıdem ve yönetim görevi ağırlıkları ile bu puan çarpılarak belirtilen oturum veya derslik için ödül puanı hesaplanır. Bu hesaplama sonucunda negatif değer oluşmaması için elde edilen ödül puanı 300.000 ile toplanır. Rulet çemberi seçimini kullanabilmek için uygunluk değerinin negatif olmaması gerekir.

“Rulet tekerleği seçiminde seçilme işlemi bireylerin uygunluk değerlerine göre yapılmaktadır. Fakat uygunluk değeri en büyük olanın seçileceği garanti edilemez, yalnız seçilme şansı daha fazla olacaktır. Bu yöntemde tüm bireylerin uygunluk değerleri bir tabloya yazılır ve toplanır. Sonra uygunluk değerleri, toplama bölünerek bireylerin [0, 1] aralığında seçilme olasılıkları belirlenir” (Nabiyev, 2005: 642). Daire şeklindeki bir rulet tekerleği bu olasılık değerlerine göre oluşturulur. Kalitesi iyi olan çözümler seçilme olasılıkları daha fazla olacağı için rulette daha fazla bir alana sahip olacaktır. Rulet tekerleği çevrilir ve durduğu nokta hangi bireyin alanına denk gelirse o birey seçilmiş olur. “Popülasyon büyüklüğüne ulaşıncaya kadar rulet tekerleği çevrilerek seçimler yapılır ve seçilen her birey eşleşme havuzu (mating pool) olarak adlandırılan bir havuzda toplanarak diğer genetik operatörlerin uygulanması için hazırlanır” (Karaboğa, 2004: 81).

Ayrıca, şubelere ayrılan dersleri veren öğretim elemanı aynı ise bu şubelerin derslerini mümkün olduğu ölçüde aynı güne ve ardışık oturumlara atanması gerekmektedir. Popülasyondaki bireylerin uygunlukları hesaplanırken, şubelere ayrılan dersler için bu kısıta uygun olan yerleşimlere 1.000 ödül puanı eklenir. Dersliklere yapılan atamalarda da, derslik kapasiteleri ile dersi alan öğrenci sayısı arasındaki farkın en yüksek olması

gerekmektedir. Her dersin atandığı derslik için bu farklar hesaplanıp toplanarak bireyin ödül puanına eklenmektedir.

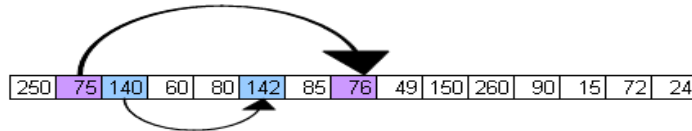
Geçerli bir ders programı oluşturmak için öncelikle katı kısıtların sağlanması gerekir. Bu nedenle ihlal edilen her katı kısıt (öğretim elemanlarının ve öğrencilerin derslerinin çakışması, her derslik türü için mevcut derslik sayısından fazla ders atanması) için “30.000” ceza puanı verilerek bu ceza puanlarının toplamı hesaplanır. Bu toplam daha önce hesaplanan ödül puanından çıkartılmak suretiyle uygunluk değerlerinin hesaplanması yapılmaktadır.

4.3. Başlangıç Popülasyonunun Oluşturulması

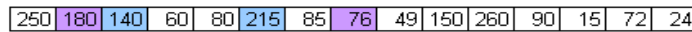
Genetik algoritmalarda diğer sezgisel arama algoritmalarında olduğu gibi elde edilebilecek sonuçların kalitesi başlangıç çözümlerine önemli ölçüde bağlılık gösterir. Bu nedenle başlangıç popülasyonundaki bireylerin yüksek kalitede olması elde edilecek sonucun daha iyi olmasını sağlar.

Literatürdeki birçok çalışmada (Özdağlar vd., 2006; Yiğit vd., 2006; Kulluk ve Türkbeş, 2004; Karaboğa, 2004) başlangıç popülasyonu oluşturulurken genellikle bireylerin rastgele olarak çözüm uzayından seçildiği ya da probleme özel sezgisel yöntemlerin kullanıldığı belirtilmektedir.

- *Başlangıç Popülasyonunu Rasgele Oluşturma (BPRO)*: İlk olarak, geliştirilen genetik algoritmada bireylerin çözüm uzayından rastgele seçilmesi yöntemi denenmiştir. Bu yöntemde her bireyin genlerinin alacağı değer, “1” ile “oturum sayısı x derslik sayısı” aralığından rastgele seçilmektedir. Her değerın seçilme olasılığı eşittir.
- *Başlangıç Popülasyonunu Rasgele Oluşturma ve Tepe Tırmanma Algoritması ile Geliştirme (BPRO-TT)*: Başlangıç popülasyonunun kalitesi yüksek olduğunda genetik algoritmalarından daha iyi sonuçlar elde edilmesi mümkündür. Bu nedenle ikinci yöntem olarak başlangıç popülasyonunu rastgele oluşturduktan sonra, popülasyondaki her bireyin tepe tırmanma algoritması kullanılarak geliştirilmesi denenmiştir. Bu şekilde oluşturulan başlangıç popülasyonu ile daha iyi sonuçlar alınacağı düşünülmüştür. Komşuluk üretme mekanizması katı kısıt ihlallerinin azaltılmasını amaçlamaktadır. Uygunluk değerlendirmesi sürecinde belirlenen ders çakışma noktalarındaki genlerin değerlerine yeni çakışmalara neden olmadan açgöz bir yaklaşımla değişim işlemi uygulanmaktadır. Eğer ihlal edilen katı kısıt kalmamışsa, açgöz bir yaklaşımla rastgele seçilen iki genin değerleri değiştirilmektedir. Genler, uygunluk değerine katkılarıyla ters orantılı bir olasılıkla değişim için seçilmektedirler. Komşu üretme mekanizması Şekil 2’de görülmektedir.



Orijinal bireyde iki nokta aynı zaman periyoduna atandığından çakışma oluşmuştur.



Bu çakışma noktalarındaki değer değiştirilerek çakışma giderilmiştir.

Şekil 2: Yerel Arama Algoritmalarında Kullanılan Komşu Üretme Mekanizması-

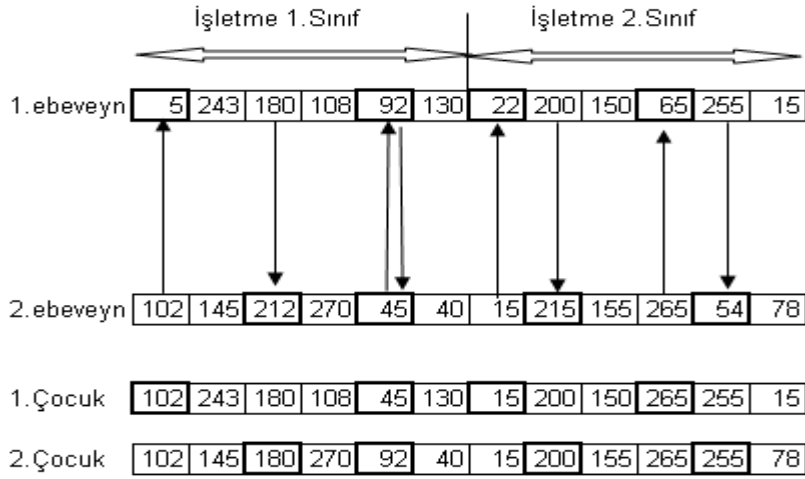
Bu şekilde katı kısıt ihlalleri nedeniyle oluşan ceza puanları azaltılacağından bireyin uygunluk değerinin artması beklenmektedir. Algoritma için iterasyon sayısı 1.000 olarak belirlenmiştir

- *Tepe Tırmanma, Tabu Arama, Tavlama Benzetimi Algoritmalarında Kullanılan Açgöz Değişim Operatörü*: Kullanılan değişim operatörü, değişimin uygulanacağı gen için verilebilecek değerlerden en yüksek puantaja sahip olanları seçmeye çalışır. Bu amaçla en yüksek puana sahip olan değerlerin bir listesi oluşturulur. Elde edilen listeden rastgele bir tanesi seçilir. Böylece yapılan değişimlerin çözümü iyileştirmede daha etkin olacağı düşünülmektedir.
- *Başlangıç Popülasyonunu Rasgele Oluşturma Tabu Arama Algoritması ile Geliştirme (BPRO-TA)*: Başlangıç popülasyonunun kalitesini arttırmak için kullanılan tabu araştırma algoritması, bireylerin uygunluk değerini arttırmayı amaçlamaktadır ve sadece kısa dönem hafıza stratejisi kullanılmıştır. Aspirasyon kriteri olarak tabu listesinde olmasına rağmen en iyi çözümden daha yüksek uygunluk değerine sahip olan hareketlere izin verilmektedir. Komşuluk üretme mekanizması tepe tırmanma algoritmasındaki gibidir.
- *Başlangıç Popülasyonunu Rasgele Oluşturma Tavlama Benzetimi Algoritması ile Geliştirme (BPRO-TB)*: Algoritma katı kısıt ihlallerini azaltarak uygunluk değerini arttırmayı amaçlar. Komşuluk üretme mekanizması tepe tırmanma algoritması ile aynı yapıdadır. Başlangıç sıcaklığı 500, sıcaklık faktörü 0.9 olarak belirlenmiştir ve sıcaklık 10 adımda bir düşürülmektedir. Komşuluk oluşturma hareketinin kabul veya reddedilmesinde Metropolis yaklaşımı kullanılmıştır.
- *Başlangıç Popülasyonunu Açgöz Rastgele Adaptif Arama Prosedürü (ARAAP) İle Oluşturma*: Araap algoritması açgöz yaklaşımı nedeniyle her gen için en yüksek puanı sağlayan alternatif değerlerden birisini rastgele seçmektedir. Bu aç göz yaklaşım neticesinde başlangıç popülasyonunu rastgele oluşturma yönteminden daha kaliteli bireyler oluşturulabileceği düşünülmektedir. Araap algoritmasının sadece açgöz rastgele oluşturma aşaması kullanılmaktadır. Yerel arama aşaması kullanılmamaktadır.

4.4. Uygun Çaprazlama Operatörünün Belirlenmesi

Genetik algoritmanın performansında çaprazlama operatörü çok etkili olabilmektedir ve farklı çaprazlama operatörleri değişik problemlerde aynı performansı göstermeyebilir. Bu nedenle permütasyon kodlama için literatürde yer alan çeşitli çaprazlama operatörleri ders çizelgeleme problemine uygulanarak karşılaştırılmıştır. Çalışmada ele alınan operatörler; tek noktali çaprazlama (one point crossover -1PX), iki noktali çaprazlama (two point crossover – 2PX), düzenli çaprazlama (uniform crossover – UX), pozisyona dayalı çaprazlama (position based crossover – PBX), sıraya dayalı çaprazlama (order based crossover – OBX), kısmi planlı çaprazlama (partially mapped crossover – PMX), çevrim çaprazlama (cycle crossover - CX), sıralı çaprazlama (ordered crossover - OX), doğrusal sıralı çaprazlama (linear order crossover – LOX), alt tur yığımları çaprazlama (sub-tour chunk crossover – SCX), en yüksek korumalı çaprazlama (maximal preservation crossover – MPX), ortada iki nokta çaprazlama (two-points centre crossover – 2PCX), sonda iki nokta çaprazlama (two-points end crossover – 2PEX), ortada ve sonda iki nokta çaprazlama (two-points end/center crossover – 2PECX) operatörleridir. Ayrıca katı ve yumuşak kısıt ihlallerini azaltmayı amaçlayan kısıt tabanlı çaprazlama - KTX operatörü geliştirilmiştir.

Kısıt Tabanlı Çaprazlama - KTX: Bu operatör çaprazlamayı üç adımda gerçekleştirir. İlk adımda, her iki ebeveyn için öğretim elemanlarının derslerindeki çakışmalar tespit edilir. Birinci çocuk için ilk ebeveynde çakışmayan dersleri temsil eden genler bu ebeveynden alınırken, çakışan dersler diğer ebeveynden alınır. Aynı işlem ikinci ebeveyn temel alınarak ikinci çocuk için tekrarlanır. İkinci adımda, öğrenci gruplarının derslerinde çakışmalar varsa bu çakışmalar aynı şekilde diğer ebeveynden alınır. Son adımda ise, aynı öğretim elemanın derslerinin farklı şubeleri aynı günde değilse bu dersi temsil eden gen diğer ebeveynden alınmaktadır. Böylece ilk adımda, öğretim elemanlarının ders çakışmaları, ikinci adımda öğrenci gruplarının ders çakışmaları ve son adımda farklı günlere atanmış şubelere ayrılmış ders sayısı azaltılmaya çalışılır. Çaprazlama sonrasında çocuklara standartlaştırma işlemi uygulanarak aynı değere sahip olan genler olması önlenir. Böylece daha iyi bireyler elde edilebilir. Çakışmaları giderecek biçimde çaprazlamanın yapılması Şekil 3’de gösterilmiştir.



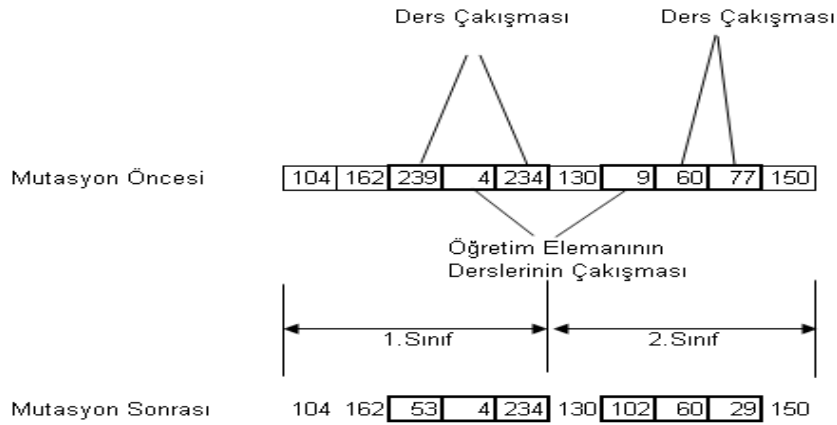
Şekil 3: KTX Operatöründe Çakışmaları Azaltma Amaçlı Çaprazlamanın Uygulanışı.

4.5. Uygun Mutasyon Operatörünün Belirlenmesi

Genetik algoritmanın performansında mutasyon operatörü de çok etkili olabilir ve farklı mutasyon operatörleri değişik problemlerde aynı performansı göstermeyebilir. Bu nedenle permütasyon kodlama için literatürde yer alan çeşitli mutasyon operatörleri ders çizelgeleme problemine uygulanarak karşılaştırılmıştır. Çalışmada ele alınan mutasyon operatörleri; tek noktada mutasyon (one point mutation -1PM), iki noktada mutasyon (two point mutation - 2PM), n noktada mutasyon (n-point mutation - NPM), iki bitişik iş takası (two operations adjacent swap - 2OAS), üç bitişik iş takası (three operations adjacent swap - 3OAS), iki rastgele iş takası (two operations random swap - 2ORS), üç rastgele iş takası (three operations random swap - 3ORS), ters çevirme mutasyonu (Inverse Mutation - IM), orta ters çevirme mutasyonu (center inverse mutation - CIM), iş kaydırma mutasyonu (shift operation mutation - SOM), geliştirilmiş rastgele iki iş takası (enhanced two operations random swap - E2ORS) operatörleridir. Ayrıca ders çizelgeleme için üç adet yerel arama tabanlı açgöz mutasyon operatörü geliştirilmiş ve denenmiştir.

- **Katı Kısıt Tabanlı Tepe Tırmanma Operatörü (KKT_TT):** Bu açgöz yaklaşım, öncelikle -katı kısıt ihlallerini gidermeye yönelik bir yerel arama yaklaşımıdır. Yerel arama algoritması olarak tepe tırmanma algoritması kullanılmaktadır. Komşu üretme mekanizması varsa katı kısıtları gidermeye yöneliktir. Her iterasyonda iki noktada

açgöz değişim uygulanmaktadır. Noktalar belirlenirken, öncelikle öğretim elemanlarının ya da öğrenci gruplarının çakışan dersleri belirlenir. Mutasyon noktası olarak bu çakışma noktalarından ikisi seçilir. Her iterasyonda öğretim elemanlarının derslerindeki çakışmalardan birisi ve öğrenci gruplarının derslerindeki çakışmalar birisi giderilmeye çalışılır. Eğer hiç ders çakışması yoksa rasgele iki mutasyon noktası seçilir ve bu iki noktada açgöz değişim uygulanır. İterasyon sayısına ulaşılmıyaya kadar bu işlemler tekrarlanır. İterasyon sayısı deneysel çalışmalar sonucu 400 olarak belirlenmiştir. Operatörün katı kısıt ihlallerini giderme mantığı Şekil 4’de gösterilmiştir.



Şekil 4: KKT_TT, KKT_TA ve KKT-TB Operatörlerinin Komşu Üretme Mekanizması.

- **Katı Kısıt Tabanlı Tabu Arama Operatörü (KKT_TA):** KKT_TT operatörünün benzeri bir operatördür. Tek farklılık kullanılan yerel arama algoritması olup bu operatörde tabu arama algoritması kullanılmaktadır. Mutasyon noktalarının belirlenmesi mekanizması aynıdır. Eğer değişimler neticesinde bireyin uygunluk değeri azalır, tabu listesi kontrol edilir ve bu iki nokta tabu listesinde yoksa değişim kabul edilir aksi halde değişim reddedilir. Aspirasyon kriteri yeni çözümün eski çözümden iyi olmasıdır. İterasyon sayısı deneysel çalışmalar sonucu 400 olarak belirlenmiştir.
- **Katı Kısıt Tabanlı Tavlama Benzetimi Operatörü (KKT_TB):** KKT_TT ve KKT_TA operatörlerinin benzeri bir operatördür. Kullanılan yerel arama algoritması tavlama benzetimi algoritmasıdır. Mutasyon noktalarının belirlenmesi mekanizması aynıdır. Eğer değişimler neticesinde bireyin uygunluk değeri azalır, Monte Carlo yöntemine göre değişimler kabul yada reddedilmektedir. Başlangıç sıcaklığı (T) 2.000.000, sıcaklık süresi (L) 5, sıcaklık faktörü (r) 0,9 ve iterasyon sayısı 400 olarak deneysel çalışmalar ile belirlenmiştir.

4.6. Uygun Üreme Operatörünün Belirlenmesi

Genetik algoritma literatüründe çaprazlama ve mutasyon işlemlerine tabi tutulmak üzere eşleştirilecek bireylerin seçilmesi için çeşitli yöntemler mevcuttur. Her yöntem farklı problemler için değişik performanslar gösterebilir. Bu nedenle geliştirilen modele en uygun seçim yöntemini belirlemek için “rulet tekerleği seçimi”, “sıralama seçimi”, “uniform seçim” ve “turnuva seçimi” yöntemleri denenmiş ve karşılaştırılarak en verimli yöntem seçilmiştir.

Rulet tekerleği seçim yöntemi kullanıldığında, karşılaşılan iki önemli problemden ilki, başlangıçta popülasyondaki uygunluğu yüksek olan bireylerin daha fazla seçilme şansı olacağından yeni nesillerde popülasyonun süper bireylerle doldurulması ve bundan dolayı erken yakınsama oluşmasıdır. İkinci önemli problem ise, nesiller ilerledikçe bireylerin uygunluk değerleri arasında farklılıkların azalması ve dolayısıyla iyi bireylerin ayırt edilememesidir. Tıkanıklık adı verilen bu durum, algoritmanın çözümü iyileştirme performansını olumsuz etkiler.

4.7. Seçkinlik (Elitizm)

Çaprazlama ve mutasyon işlemleri neticesinde o ana kadar bulunmuş en iyi bireyin kaybedilme riski mevcuttur. Bu nedenle en iyi çözümü korumak amacıyla seçkinlik yaklaşımları uygulanır. Seçkinlik işlemi iki farklı şekilde uygulanabilir. Birincisi, o ana kadar bulunan en iyi çözümün yeni nesile kopyalanması şeklindedir. İkincisi ise, çaprazlama ve mutasyon işlemlerinde seçkinlik yaklaşımıdır. Çaprazlama operatöründe ebeveynler ve elde edilen çocuklardan en iyi ikisi yeni nesile aktarılır. Mutasyon operatöründe ise sadece çözümü geliştiren değişimlere izin verilir. Her iki yöntemde denenmiş ve değerlendirilmiştir.

4.8. Optimal Konfigürasyon

- Çaprazlama Operatörü: Katı kısıt tabanlı çaprazlama (KTX), ortada ya da sonda iki nokta çaprazlama (2PECX), tek noktada çaprazlama (1PX) operatörleri üst sezgisel tarafından yönetilmektedir.
- Mutasyon Operatörü: Katı kısıt tabanlı tabu arama (KKT_TA), katı kısıt tabanlı tepe tırmanma (KKT_TT) ve katı kısıt tabanlı tavlama benzetimi (KKT_TB) operatörleri üst sezgisel tarafından yönetilmektedir.
- Sezgisel (Operatör) Seçim Yöntemi: Hareketli Ortalama Seçim Fonksiyonu
- Popülasyon Büyüklüğü: 6
- Çaprazlama Oranı: % 60
- Mutasyon Oranı: % 5
- Adaptif Mutasyon Stratejisi: n = 10.000; d = % 5; lmt = % 30
- Başlangıç Popülasyonu: ARAAP-TB
- Seçkinlik Stratejisi: “Operatör seçkinliği ve en iyi bireyin korunması” yöntemi.
- Seçim Yöntemi: T = 5.000.000 olan Boltzman Seçimi, w=2 olan pencere ölçekli (ölçekleme penceresi) rulet tekerleği seçimi ve c=1,2 olan doğrusal ölçekli rulet tekerleği seçimi operatörleri üst sezgisel tarafından yönetilmektedir.
- Popülasyon Yenileme Süresi: 15 saniye.

4.9. Yazılımın Test Edilmesi

İdeal konfigürasyon tespit edildikten sonra Celal Bayar Üniversitesi İktisadi ve İdari Bilimler Fakültesi Öğrenci İşleri Şefliği’nden 2007-2008 öğrenim yılı bahar yarıyılına ait veriler alınmış ve programa girilerek kaydedilmiştir. Öğretim elemanlarının tercih ettikleri oturumlar ve dersliklere ilişkin kısıtlamalar girilmiş ve çözüm algoritması test edilmiştir.

2007-2008 öğrenim yılı bahar yarıyılında 3 bölüm ve 4 sınıfa ait, şubelere ayrılan 55, ayrılmayan 39 ders olmak üzere toplam 94 dersin oturumlara atanması gerekmektedir. Derslerin atanabileceği oturum sayısı 40, derslik sayısı 27’dir. Dersliklerin 2’si bilgisayar laboratuvarı, 3’ü projeksiyonlu derslik, 4’ü amfi ve 18’i normal dersliklerdir. Dersleri

atanacak öğretim elemanlarının sayısı 51'dir. Öğretim elemanlarının 45'i oturum tercihi yaparken 20'si derslik tercihinde de bulunmuştur.

Gerçek verilerle yapılan testlerin sonuçlarını değerlendirebilmek amacıyla öğrenci işleri şefinin elle düzenlediği ders programı incelenmiştir ve tüm katı ve yumuşak kısıtların sağlanmış olduğu görülmüştür. Bu ders programına göre elde edilebilecek en yüksek uygunluk değeri 3.840.395 olarak hesaplanmıştır.

Gerçek verilerle 0.5 , 1, 5, 10, 20, 30 ve 60 dakikalık süreler için ve 1,2 ve 3 koşum için performans testleri yapılmıştır. Yapılan testlerin sonuçları Tablo 4'de gösterilmiştir. Tablo 4 incelenirse, algoritmanın 0.5 dakika süre ile 1 koşum çalıştırılması halinde bile öğretim elemanlarının derslik ve oturum tercihlerinin % 87'si yerine getirilmektedir ve herhangi bir zorunlu kısıt ihlal edilmemektedir. Şubelere ayrılmış olan ve öğretim elemanı aynı olan derslerin % 39'u kısıta uygun olarak yerleştirilmiştir. Çalışma süresi arttırıldığında tercihlerin ve şubelere ayrılmış derslere ilişkin kısıtların daha büyük oranda sağlandığı görülebilir. Algoritmanın 20 dakikalık sürede 1 koşum çalıştırılmasında, 10 dakikalık sürede 2 ve 3 koşum çalıştırılmasında öğretim görevlilerinin derslik ve oturum tercihlerinin tamamı sağlanabilmiştir. Algoritmanın 60 dakikalık süre ile 2 koşum çalıştırılmasında elde edilen sonuca bakıldığında şubelere ayrılmış derslere ilişkin kısıtın ve öğretim elemanlarının tercihlerinin tamamının sağlandığı görülmektedir. Ancak, derslik kapasitelerine ilişkin yumuşak kısıt nedeniyle en iyi sonuç elde edilememiştir. Algoritma, 60 dakika süre ile 3 koşum çalıştırıldığında ise en iyi çözüme ulaşılabilmektedir. Sezgisel algoritmalarda elbette her zaman en iyi çözüme ulaşmak mümkün olmayabilir. Bu çalışmada geliştirilen genetik algoritma da optimal çözümü garanti etmemektedir (Palamutçuoğlu, 2008: 174). Yapılan denemeler sonucunda, algoritmanın çalışma süresinin mümkün olduğunca uzun tutulmasının ve koşum sayısının mümkün olduğunca fazla olmasının daha iyi çözümler elde edilmesini sağladığı görülmüştür. Tavsiye edilen çalışma süresi en az 60 dakika ve koşum sayısının da en az 3 olmasıdır.

Algoritmanın 60 dakikalık 3 koşum çalıştırılması ile elde edilen bu ders programı, elle yapılan ders programı ile karşılaştırıldığında, derslerin atandığı oturumlarda ve dersliklerde farklılıklar olduğu gözlenmiştir. Oturumlardaki farklılık, öğretim elemanlarının oturum tercihlerinin geniş olmasından kaynaklanmaktadır. Bu nedenle yapılan her iki ders programı da öğretim elemanları açısından tatmin edici olmuştur. Derslik atamaları incelendiğinde ise program tarafından yapılan atamaların daha tatminkâr olduğu söylenebilir. Dersleri alan öğrencilerin sayısı derslik kapasitesini aşmayacak ya da en az ölçüde aşacak biçimde ders çizelgesi oluşturulmuştur.

1. Koşum							
Süre	0,5 dakika	1 dak.	5 dak.	10 dak.	20 dak.	30 dak.	60 dak.
Uygunluk	3.799.834	3.818.280	3.852.460	3.827.552	3.831.752	3.832.956	3.836.956
Tercih puanı	3.769.795	3.771.275	3.771.975	3.772.125	3.772.395	3.772.395	3.772.395
Aynı güne atanamayan şube dersleri sayısı	19	11	8	7	5	4	2
Derslik sayılarını aşan atamaların sayısı	0	0	0	0	0	0	0
Öğretim elemanlarının derslerinde kaçışma sayısı	0	0	0	0	0	0	0
Öğrencilerin derslerinde kaçışma sayısı	0	0	0	0	0	0	0
Öğretim elemanlarının yerine getirilmeyen tercihlerinin sayısı	6	4	2	1	0	0	0
Derslik kapasiteleri ile öğrenci sayılarının arasındaki fark	6.039	7.005	7.485	7.427	7.357	6.561	6.561
2. Koşum							
Süre	0,5 dakika	1 dak.	5 dak.	10 dak.	20 dak.	30 dak.	60 dak.
Uygunluk	3.812.716	3.821.266	3.826.440	3.827.726	3.834.988	3.838.434	3.840.407
Tercih puanı	3.770.345	3.771.695	3.772.075	3.772.395	3.772.395	3.772.395	3.772.395
Aynı güne atanamayan şube dersleri sayısı	15	10	7	4	3	1	0
Derslik sayılarını aşan atamaların sayısı	0	0	0	0	0	0	0
Öğretim elemanlarının derslerinde kaçışma sayısı	0	0	0	0	0	0	0
Öğrencilerin derslerinde kaçışma sayısı	0	0	0	0	0	0	0
Öğretim elemanlarının yerine getirilmeyen tercihlerinin sayısı	5	3	1	0	0	0	0
Derslik kapasiteleri ile öğrenci sayılarının arasındaki fark	6.571	6.571	7.365	7.351	7.723	6.039	6.012
3. Koşum							
Süre	0,5 dakika	1 dak.	5 dak.	10 dak.	20 dak.	30 dak.	60 dak.
Uygunluk	3.813.600	3.824.622	3.829.056	3.833.208	3.835.366	3.840.956	3.840.956
Tercih puanı	3.771.285	3.772.015	3.772.145	3.772.395	3.772.395	3.772.395	3.772.395
Aynı güne atanamayan şube dersleri sayısı	11	8	6	3	2	0	0
Derslik sayılarını aşan atamaların sayısı	0	0	0	0	0	0	0
Öğretim elemanlarının derslerinde kaçışma sayısı	0	0	0	0	0	0	0
Öğrencilerin derslerinde kaçışma sayısı	0	0	0	0	0	0	0
Öğretim elemanlarının yerine getirilmeyen tercihlerinin sayısı	4	2	1	0	0	0	0
Derslik kapasiteleri ile öğrenci sayılarının arasındaki fark	6.425	7.027	7.261	7.213	7.691	6.561	6.561

Tablo 4: Gerçek Veriler Kullanılarak Yapılan Performans Testlerinin Sonuçları.

SONUÇ

Yazılımın çözüm algoritmasında kullanılan yaklaşım, melez genetik algoritmadır. Algoritmanın başlangıç çözümleri (popülasyon) oluşturulurken çeşitli yaklaşımlar denenmiş ve uygun olan yaklaşım deneysel olarak belirlenmiştir. Algoritmada, permütasyon tipi kodlama şeması kullanılmıştır. Başlangıç çözümün oluşturulmasında açgöz rasgele adaptif arama prosedürü uygulanmış ve yerel arama aşaması içinse katı kısıt odaklı bir tavlama benzetimi algoritması kullanılmıştır. Seçim yöntemi olarak rulet tekerleği seçimi, sıralama seçimi, turnuva seçimi ve uniform seçim yöntemleri denenmiştir. Ayrıca rulet tekerleği seçiminde, çeşitli ölçekleme yöntemleri de denenerek uygun seçim yöntemi belirlenmiştir. Sonuçta rulet tekerleği yönteminin ölçeklenmiş bir türü olan boltzman seçiminin kullanılması uygun görülmüştür. Genetik algoritmanın performansında önemli etkiye sahip olabilen çaprazlama ve mutasyon operatörleri de deneysel çalışma neticesinde belirlenmiştir. Bu amaçla literatürde permütasyon kodlama için kullanılan çeşitli çaprazlama ve mutasyon operatörleri programlanmış ve denenmiştir.

Literatürde yer alan bu operatörlere ek olarak 1 adet çaprazlama (KTX) ve 3 adet mutasyon operatörü (KKT_TT, KKT_TA, KKT_TB) geliştirilmiş ve denenmiştir. En uygun çaprazlama operatörü olarak KTX operatörü, mutasyon operatörü olarak da KKT_TA operatörü seçilmiştir. Çaprazlama ve mutasyon operatörleri nedeniyle elde edilen en iyi çözümün kaybedilmesi, ve popülasyon kalitesinin azalmasını önlemek amacıyla seçkinlik işlemi uygulanmaktadır. Uygun seçkinlik yaklaşımı deneysel olarak tespit edilmiştir. Algoritmanın performansı için önemli etkilere sahip olan çaprazlama oranı, mutasyon oranı ve popülasyon büyüklükleri deneysel olarak tespit edilmiştir. Seçilen popülasyon büyüklüğü 6 olduğu için, popülasyondaki çeşitliliğin devamını sağlamak amacıyla belirli zaman aralığıyla popülasyon yenilenmektedir. Popülasyon yenilenirken o ana kadar elde edilen en iyi çözüm saklanmaktadır. Böylece en iyi çözümün kaybedilmesi riski ortadan kaldırılmaktadır. Ayrıca popülasyon çeşitliğini sağlamak ve yerel optimumdan kaçabilmek amacıyla adaptif bir mutasyon stratejisi kullanılmaktadır. Popülasyon standart sapması belirli bir değerin altına düştüğünde mutasyon oranı artırılarak algoritma yerel optimumdan kaçmaya çalışır. Bu strateji sayesinde algoritma popülasyon çeşitliliğini sağlamaya ve erken yakınsamayı önlemeye çalışmaktadır.

Geliştirilen algoritmada alınan önlemlere rağmen algoritmanın her zaman en iyi çözümü bulması mümkün olmayabilir. Bu nedenle kullanıcıya algoritmanın birkaç kere koşturulması imkanı sunulmuştur. Böylece en iyi çözümün elde edilmesi olasılığı daha fazla olmaktadır. Algoritma yaklaşık 30 saniyede olurlu ve yüksek kalitede bir çizelge oluşturabilmektedir. Ancak, çalışma süresi ve koşum sayısı ne kadar fazla olursa elde edilecek çözümler daha kaliteli olacaktır. Çalışma süresinin en az 30 ya da 60 dakika olması, popülasyonun 15 saniyede bir yenilenmesi ve algoritmanın en az iki defa koşturulması iyi sonuçlar verecektir.

Oluşturulan yazılım ile ders programı oluşturmanın ders programının elle yapılmasına göre çeşitli avantajları görülmüştür. Bu avantajlar:

- Ders programları için ilgili personelin 2-3 gün zaman harcadığı düşünüldüğünde, çözümün 2-3 saat gibi bir sürede elde edilebilmesinin eğitim kurumları için önemli bir avantaj olacağı düşünülebilir.
- Oluşturulan bir ders programında bazı derslerin yerleşiminde değişiklikler yapılması gerekebilir. Böyle bir durumda yazılım, seçilen ders için uygun olan oturum ve derslikleri gösterir ve dersin belirlenen yeni oturum ve dersliğe atanabilmesini sağlar.

Bu da kullanıcılara önemli ölçüde bir kolaylık sağlamaktadır. Bir dersin herhangi bir oturum ve dersliğe atanması durumunda çizelgede yapılması gereken değişiklikleri göstermektedir. Onay verilmesi durumunda da bu değişiklikleri gerçekleştirmektedir. Bu iki alternatif yolla oluşturulmuş bir çizelge 1-2 saniye gibi bir zamanda yeniden çizelgelenebilmektedir.

■ Yazılım, çözüm uzayının çeşitli bölgelerini taradığı için personelin elle yaptığı ders programından daha kaliteli programlar oluşturabilmektedir.

■ Ders programı yapıldıktan sonra, programın çeşitli kriterlere göre raporlanması gerekmektedir. Bu raporlama işleminin personel tarafından yapılması da uzun bir zaman almaktadır. Yazılım ile oluşturulan ders programının çeşitli kriterlere göre raporları oluşturabilmesi de önemli bir avantaj olarak görülmektedir.

Sonuçları genel olarak değerlendirildiğinde, problemin kısıtlarına ve amacına özel olarak geliştirilen operatörlerin daha başarılı olabildikleri, başlangıç popülasyonunun oluşturulmasında sezgisel yöntemlerin kullanılmasının genetik algoritma performansını olumlu yönde etkilediği, yerel arama tabanlı sezgisel mutasyon operatörlerinin daha etkili olduğu, adaptif stratejilerin olumlu sonuç verdiği söylenebilir.

Çok işlemci desteği olan bilgisayarlarda geliştirilen genetik algoritmaların çok kanallı paralel genetik algoritma yapısına dönüştürülmesi düşünülebilir. Çok kanallı bir yazılım geliştirilerek her kanalda bir genetik algoritmanın çalıştırılması ve zaman zaman bu genetik algoritmaların elde ettikleri sonuçlar hakkında haberleşmeleri şeklinde bir model geliştirilebilir. Paralel bilgi işleme ve çok sayıda işlemcinin kullanılması nedeniyle geliştirilen algoritmadan daha kısa sürede daha etkin sonuçlar alınabilmesi mümkündür. Bu uygulama kapsamından paralel genetik algoritma yaklaşımı denenmiştir, ancak çok sayıda popülasyonun gelişimi için bütün işlemler tek bir işlemci ile sürdürüldüğünden yeterli performans sağlanamamıştır. Çok işlemcili bir bilgisayarda paralel genetik algoritmanın daha başarılı olacağı düşünülmektedir.

Paralel genetik algoritma modeline benzer fakat her kanalda farklı bir sezgisel üstü (meta sezgisel) yöntemin çalıştırıldığı çok kanallı bir uygulama modeli de geliştirilebilir. Örneğin bir kanalda genetik algoritma çalışırken, diğer bir kanalda yapay karınca sistemi ve belki üçüncü bir kanalda yapay bağımsızlık sistemi gibi algoritmalar çalıştırılabilir. Her kanalda elde edilen en iyi çözümler kanallar arasında paylaşılacağı için üç farklı sezgisel yöntemin avantajlarından faydalanma imkânı söz konusu olabilir.

KAYNAKLAR

- Carter, M. W. ve Laporte, G. , 1998, "Recent Developments in Practical Course Timetabling", In Selected Papers From the Second international Conference on Practice and Theory of Automated Timetabling II (August 20 - 22, 1997), Burke, E. K. ve Carter, M. W., Eds. Lecture Notes In Computer Science, (1408); 3-19, Springer-Verlag.
- Glover, F. (1989), "Tabu Search – Part I", ORSA Journal on Computing 1; 190-206.
- Glover, F. (1990), "Tabu Search – Part II", ORSA Journal on Computing 2; 4-32.
- Glover, F. and M. Laguna (1993), "Tabu Search," Modern Heuristic Techniques for Combinatorial Problems, C. Reeves, ed., Blackwell Scientific Publishing; 70-141.

- Grefenstette, J.J. (1986), "Optimization of Control Parameters for Genetic Algorithms," IEEE Trans. on Systems, Man, and Cyb., (16), No. 1; 122-128.
- Karaboğa, D. (2004), Yapay Zeka Optimizasyon Algoritmaları, İstanbul, Atlas Yayın Dağıtım.
- Kirkpatrick, S. Gelatt, D. ve Vecchi, M. P. (1983), "Optimization by simulated annealing", Science, (220), No.4598; 671-680.
- Kulluk, S., Türkbey, O., (Haz.15-18, 2004), "Tesis yerleşim problemleri için bir genetik algoritma", YAEM'2004-Yöneylem Araştırması ve Endüstri Mühendisliği XXIV. Ulusal Kongresi, Çukurova Üniversitesi, Gaziantep Üniversitesi Endüstri Mühendisliği Bölümleri ve Yöneylem Araştırması Derneği, Bildiriler kitabı; 503-505, Adana, Türkiye.
- Metropolis, N. , Rosenbluth, A.W., Rosenbluth, M.N., Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines", Journal of Chemical Physics, (21), No.6; 1087-1092.
- Nabiyev, V.V. (2005) , Yapay Zeka Problemler - Yöntemler – Algoritma, Ankara, Seçkin Yayıncılık.
- Nehi, H. M. Gelareh, S. (2007), "A survey of meta-heuristic solution methods for the quadratic assignment problem", Appl. Math. Sci.Online Edition, (1), No.46; 2293-2312.
- Özdağlar, D., Benzedem, E. Ve Kahraman A.M. (2006), "Kompleks Su Dağıtım Şebekelerinin Genetik Algoritma ile Optimizasyonu", İMO Teknik Dergi, (253); 3851 -3867.
- Palamutçuoğlu, B. T. (2008), Üretim Ve Hizmet Planlamasında Çizelgeleme Problemlerinin Yöneylem Teknikleriyle Çözümü : Ders Ve Sınav Programlarının Optimizasyonu Üzerine Bir Uygulama, Yüksek Lisans Tezi, Celal Bayar Üniversitesi, Sosyal Bilimler Enstitüsü, Manisa.
- Schaerf, A., 1995, "A Survey of Automated Timetabling (Technical Report CS-R9567)" , Amsterdam, Centrum voor Wiskunde en Informatika-CWI, The Netherlands.
- Yiğit, V., Aydın, M. E., Türkbey, O., (2006), "Solving large-scale uncapacitated facility location problems with evolutionary simulated annealing", International Journal of Production Research, (44), No. 22; 4773-4791.
- Yiğit, V., Türkbey, O. (2003), "Tesis Yerleşim Problemlerine Sezgisel Metodlarla Yaklaşım", Gazi Üniv.Müh.Mim.Fak.Dergisi, (18), No.4; 45-56.