

SİPARIŞ TİPİ ATÖLYELERDE İŞ SIRALAMA PROBLEMİ İÇİN BİR GENETİK ALGORİTMA UYGULAMASI

Murat Baskak

İ.T.Ü. İşletme Fakültesi, Endüstri Mühendisliği Bölümü, 34367 Maçka/İstanbul

Vural Erol

Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Sistem Mühendisliği Programı, 34357 Beşiktaş/İstanbul

ÖZET

Üretim sistemlerinde iş sıralama problemi, belli bir performans ölçütünü eniyileyecek iş akış sırasını (tezgâhlara gelen işlerin yapılma sırasını) belirlemek olarak bilinir. İş sıralama problemleri, işlerin geliş şekline göre Akış tipi ve Sipariş tipi olmak üzere ikiye ayrılır. Akış tipi iş sıralama problemlerinde, belirli bir dönem için iş listesi bilinmektedir ve tüm işlerin makinalarda işlenme sırası aynıdır. Sipariş tipi problemlerde ise iş listesi sürekli ve rastgele değişmekte, işler düzensiz aralıklarla atölyeye gelmektedir. Burada en önemli nokta ise işlerin makinalarda işlenme sıralarının farklı olabilmesidir. Bu tip problemlerde genel amaç, genellikle tüm işlerin tamamlanma süresini enküçükmektir. Bu durumu gerçekleştirmek için kullanılan en etkili yöntemlerden biri de Genetik Algoritmalar'dır (GA). GA, olası tüm çözümlerin değil de salt bazı seçilmiş çözümlerin denenmesi yoluyla beklenen optimum sonucu bulmaya çalışan, parametre kodlama temeline dayanan bir arama tekniğidir. GA'lar doğada geçerli olan "en iyinin yaşaması" kuralına dayanarak sürekli iyileşen çözümler üretir. Bunun için "iyi" nin ne olduğunu belirleyen bir uygunluk fonksiyonu ve yeni çözümler üretmek için yeniden kopyalama ve değiştirme gibi operatörleri kullanır. GA'lar, "Yapay Zekâ" nın gittikçe genişleyen bir kolu olan evrimsel hesaplama tekniğinin bir parçasını oluşturmaktadır. Bu bildiride, GA yöntemiyle Sipariş tipi iş sıralama problemlerinin çözümü incelenmekte, bu konuda Visual Basic.Net'te hazırlanan bir program kullanılarak bir elektrik motoru fabrikasında yapılan uygulamanın sonuçları değerlendirilmektedir.

Anahtar Kelimeler: Üretim Plânlama, Çizelgeleme, Yapay Zekâ, Uzman Sistemler

ABSTRACT

In production systems, it is known that job sequencing problem is the determination of work flow sequence (jobs that come to workbenches for operating) which improves a certain performance criterion. There are two kinds of job sequencing problem according to work arrival condition, as Flow type and Order type. In Flow type job sequencing problems, work list for a specific period is known and all of the work operating sequences on machines are same. As for Order type job sequencing problems, work lists can be randomly changeable and jobs comes to workshops unperiodically. Order type job sequencing problem's the most important feature is that work operating sequences on machines can be different. The general objective in these type problems is usually minimizing the completion time of all works. For realizing this situation, Genetic Algorithms (GA) is one of the most effective methods. GA is a search method based on parameter encoding that purposes optimum outcome using some of chosen solutions. GA produces continually improving solutions on the basis of nature rule which provides "living which has best properties". Because of this, it uses fitness function that determines "best properties" and cross over operator for producing new solutions. GA is an evolutionary calculation technique which expands with Artificial Intelligence. In this article, problem solutions in Order type job sequencing with GA is examined and application results from a electrical motor factory, by using a program which is prepared in Visual Basic.Net are evaluated.

Keywords: Production Planning, Schedullng, Artificial Intelligence, Expert Systems

1. GİRİŞ

Günümüzün rekâbete dayalı ortamında, işletmeler en az miktarda kaynak kullanarak, müşteri gereksinimlerine en hızlı yanıt verebilecek tekniklerle en kaliteli ürün ve hizmet üretmenin peşindedirler. Bu yüzden, hızlı değişen müşteri talepleri karşısında özellikle üretim plânlarını en çabuk oluşturan işletmeler rekâbette bir adım öne geçecektir. Etkili plânlar kurabilmek için de işletmeler optimizasyon tekniklerinden yararlanmak zorundadırlar. Klâsik analitik yöntemler, şimdiye kadar plânların oluşturulmasında en çok kullanılan yöntemlerden biri idi. Ama, üretim sistemlerinde bulunan çok fazla rassal parametre ve sürekli değişen çevre, kurulması istenen bu plânların yavaş oluşturulmasına ve etkisizleşmesine neden olabilmektedir. Bu nedenle, çok değişkenli sistemlerde oldukça etkili olan ve gelişimini hâlâ sürdüren bir optimizasyon tekniği olan "Genetik Algoritma" yöntemi, araştırmacılar tarafından bu plânların oluşturulmasında kullanılmaya başlanmıştır.

Endüstri Mühendisliği'nin temel hedefi, sistemi en verimli duruma getirmektir. Bu da optimum kaynak kullanımıyla gerçekleştirilebilir. Bu noktada, Genetik Algoritmaların çalışma hızı, sunduğu istenilene yakın çözümler ve esneklik, diğer bir deyişle tüm sistemlere uygulanabilir olması nedeniyle Endüstri Mühendislerinin en çok kullandığı tekniklerden biri olacaktır.

Bu çalışmada, bir Endüstri Mühendisliği konusu olan İş Sıralama üzerinde Genetik Algoritma yöntemi incelenmektedir ve bu yöntemle Sipariş Tipi Atölyelerde İş Sıralama problemine çözümler sunan bir programla uygulama yapılmaktadır. Bu bildiride ikinci bölümde, Sipariş Tipi Atölyelerde İş Sıralama problemi tanımlanıp ayrıntılarıyla incelenmektedir. Üçüncü bölümde, Genetik Algoritma tekniği ana hatlarıyla irdelenmektedir. Dördüncü bölümde ise İş Sıralama konusunda yapılan Genetik Algoritma çalışmaları sıralanıp, bu projenin uygulama kısmında kullanılan teknikler ayrıntılı olarak açıklanmaktadır. Beşinci bölümde, Visual Basic.Net'te yazılan, Genetik Algoritma yöntemini kullanarak Sipariş Tipi Atölyelerde İş Sıralama yapan programın nasıl çalıştığı anlatılmaktadır. Altıncı bölümde ise, Arçelik Çok Amaçlı Motor Fabrikası'ndan alınan verilerle, programı kullanarak Sipariş Tipi Atölyelerde optimum iş sırası bulunmaya çalışılıp, sonuçlar yorumlanmaktadır.

2. SİPÂRİŞ TİPİ ATÖLYELERDE İŞ SIRALAMA PROBLEMİ

2.1. Genel Bilgiler

İş Sıralama konusuna geçmeden önce, konunun içinde yer aldığı Üretim Plânlama hakkında genel bilgiler vermek yararlı olacaktır.

Üretim Plânlama, üretilecek ürünü belirlemek, üretim için donanım gereksinimlerini saptamak ve ürünlerin istenen kalite ve mâliyetle, istenen sürede, doğru zamanlarda ve istenen miktarlarda oluşumunu sağlayacak programlama ve çizelgeleme faaliyetlerinden oluşan bir üretim yönetimi etkinliğidir. Üretim Plânlama, işletmelerin temel amaçları olan; toplam kârın enbüyüklemesi, siparişlerin zamanında karşılanması, eldeki kaynakların (makina, işgücü, donanım, malzeme vb.) en verimli şekilde kullanılması gibi amaçlara önemli ölçüde katkıda bulunur (Acar, 2001).

Üretim Plânlama adımları genel olarak şunlardır:

1. Satış Tahminleri Yapma
2. Toplu Üretim Plânlama ve Ana Üretim Programlama
3. Malzeme Gereksinim Plânlaması
4. Üretim Programlama (Ayrıntılı Plânlama)

Üretim Programlamada, ana üretim programının belirlediği politikalar çerçevesinde yapılacak işlerin sırası ve bu işlere ayrılacak optimum kaynak miktarı belirlenir. Üretim programlamanın son aşamasında haftalık iş çizelgeleri hazırlanır. İş çizelgelerinde operasyonlar kaynaklara zaman boyutu gözönüne alınarak aktarılır. Konumuz olan Sipariş Tipi Atölyelerde İş Sıralama problemi de bu adımda karşımıza çıkmaktadır.

2.2. İş Sıralama

Aynı tezgâhtan/tezgâhlardan geçecek birden çok iş olduğu takdirde, bu işlerin yapılma sırasının belirlenmesi işlemine İş Sıralama denir. İşlerin her tezgâhta hangi zamanda başlaması ve bitmesi gerektiğinin plânlanıp gösterildiği çizelgelere iş çizelgesi, bu plânlama çalışmasına, yâni elde edilen iş sıralarına zaman boyutunun katılması işlemine iş çizelgeleme denir (Acar, 2001).

Gerçekte sıralama, çözülmesi genellikle güç olan karmaşık bir sorundur. Örneğin, bir işlem noktasında sıra bekleyen on iş; 1-2-3-4-5-6-7-8-9-10, 2-1-3-4-5-6-7-8-9-10, ..., 10-9-8-7-6-5-4-3-2-1 biçiminde (10!=) 3.628.800 tane değişik sırada işlemde geçebilir.

Sıralama için kullanılan öncelik kurallarından (bekleyen işler için işi başlatma kuralları) bazıları aşağıda verilmiştir:

1. İlk gelen ilk yapılır.
2. Son gelen ilk yapılır.
3. İşlem süresi en kısa olan ilk yapılır.
4. İşlem süresi en uzun olan ilk yapılır.
5. Teslim tarihi en önce olan ilk yapılır.
6. Gevşek süresi en az olan ilk yapılır.
7. Rastgele seçim.
8. En yüksek mâliyetli olan ilk yapılır.

İşi başlatma kuralları, endüstride yaygın olarak kullanılmakta olup; işi başlatmanın etkinliği, büyük ölçüde tüm üretimin etkinliğini belirler.

Üretim sistemlerinde sıralama problemi, belli bir performans ölçütünü eniyileyecek iş akış sırasını (tezgâhlara gelen işlerin yapılma sırasını) belirlemek olarak bilinir. Klâsik İş Sıralama problemi işlerin geliş şekline göre iki türe ayrılır (Acar, 2001):

a. **Akış Tipi Atölyelerde İş Sıralama:** Bu tip problemlerde, belirli bir dönem için iş listesi bilinmektedir. Tüm işlerin makinalarda işlenme sırası aynıdır. İşler, boş olan bir atölyeye hemen işlenmek üzere düzenli olarak gelirler. Bu tip problemlerde genel amaç, işlerin tezgâhlardan geçiş sırasını bulmaktır.

b. **Sipariş Tipi Atölyelerde İş Sıralama:** Bu tip problemlerde ise iş listesi sürekli ve rastgele değişmekte, işler düzensiz aralıklarla atölyeye gelmektedir. Burada en önemli nokta ise işlerin makinalarda işlenme sıralarının farklı olabilmesidir. Bu tip problemlerde genel amaç, tüm işlerin işlenme süresini (yayıma süresi) enküçüklemek, tezgâh boş bekleme süresini veya bekleyen iş sayısını enazlamak vb. olabilir.

Diğer optimizasyon problemlerinde olduğu gibi, işlerin sıralanmasında da toplam etkinliğin eniyilenmesi istenir. Atölye performansını değerlendirmek için kullanılan ölçüt, sıralama probleminde önemli rol oynamaktadır. Etkinliğin eniyilenmesinde seçilebilecek kimi ölçütler şunlardır:

- Tüm işlerin tamamlanma zamanının (İş tamamlanma zamanlarının en büyüğü veya en son sırada yapılan işin bitiş zamanı veya yayılma süresi) enküçüklenmesi
- İş sırasının ortaya çıkardığı ortalama bekleme süresinin enküçüklenmesi
- Kuyrukta bekleyen ortalama iş sayısının yansıttığı üretim içi stok düzeyi (geciken iş sayısı, yapılmış ürün miktarı) değerinin enküçüklenmesi
- Tezgâh ve işgücü kullanım oranlarının enbüyüklenmesi

Görüldüğü gibi ölçütler içinde geçen temel değerler; akış süresi, gecikme süresi, geciken iş sayısı gibi değerlerdir. Ama bu tip problemlerde en önemli amaç, sistemin yayılma süresini enküçüklemektir. Örneğin P_{ij} : j tezgâhında yapılacak i işinin işlem süresini ve C_{ij} : j tezgâhında yapılan i işinin tamamlanma zamanını gösteriyorsa, m; kullanılan tezgâh sayısı olmak üzere, örneğin yayılma süresi şu şekilde hesaplanır:

$$M = \text{Enb } \{C_{ij} \mid (i=1,2,\dots,n), (j=1,2,\dots,m)\} \quad (1)$$

2.3. İş Sıralama ile İlgili Yöntemler

İş Sıralama problemlerinde amaç fonksiyonu, genellikle toplam akış süresinin ağırlığını ve en son işin tamamlanma zamanını (makespan) enküçüklemektir. İş çizelgelemede; F toplam akış süresini ve F^l ortalama akış süresini enküçüklemek, NP (Non-Polinomial) problemler kapsamına girer (Gen ve Cheng, 2000). Genel olarak iş çizelgelemede kullanılan yöntemler aşağıda verilmiştir:

1. Matematiksel programlama
2. Öncelik kurallarına göre sıralama
3. Simülasyon tabanlı yöntemler
4. Yapay zekâ tabanlı yöntemler
5. Sezgisel yöntemler
6. Çok temsilcili paradigmlar
7. Üretim düzgünleştirme operatörü yöntemi

Çizelgeleme optimizasyonunda matematiksel programlama kullanmak, hesaplama süresinin çok uzun olması nedeniyle uygulanabilir değildir. Simülasyon tabanlı çizelgeleme programları, esnek olduğundan tercih edilmektedir. Konumuz olan Genetik Algoritmalar ise yapay zekâ tabanlı çizelgeleme yöntemleri arasında son zamanlarda önemli bir araştırma konusu olmuştur.

2.4. Sipariş Tipi Atölyelerde İş Sıralama

Sipariş Tipi Atölyelerde İş Sıralama probleminin çözüm uzayı oldukça geniştir. Bu nedenle yöneylem modelleriyle en az yayılma süresini verecek iş sırasını bulmak oldukça zor ve zaman alıcıdır. Örneğin, 8 iş ve 4 makinanın bulunduğu bir sistemin çözüm uzayı $[(8!)^4 =] 2,64 \cdot 10^{18}$ elemanlıdır. Bu nedenle iş listesinin sürekli ve rastgele değiştiği Sipariş Tipi Atölyelerde (değişken) sıralama problemlerinde çözüme ulaşmak için tümüyle farklı teknikler kullanmak gerekir.

Bu tekniklerden biri de öncelik kurallarına göre sıralama yapmaktır. Öncelik kuralları, herhangi bir tezgâhta, belli bir işin bitiminden sonra hangi işe başlanacağı konusunda karar verici kişiye yol gösteren basit mantıksal kurallardır. İşlerin özellikleri ya da sistemin durumuna göre işlere öncelik değerleri vermede kullanılırlar. Örneğin, bir parçanın üretimi için gerekli işlem sayısı, parçanın kalan işlem süresi vb. Ama bu teknik, %100 güvenilir olmadığı gibi, gerçekte elde edilebilecek en iyi sonuca çok uzak çözümler sunabilmektedir.

İş Sıralama, iyi bilinen zor kombinatoriyal optimizasyon problemlerinden biridir. Son otuz yılda bu problemlerin çözümü için yeteri kadar araştırma yapılmış olup, öncelik kurallı ve dal-sınır tekniğine dayanan çeşitli yöntemler geliştirilmiştir. Bu tip problemler, NP türü problemler kapsamında olduğundan bilgisayarların hızlarının artması ile daha etkin çizelgeleme yöntemleri geliştirilmeye çalışılmıştır (Baker, 1974). Genetik Algoritmalar da Sipariş Tipi Atölyelerde İş Sıralama problemini çözmek için süre ve alınan sonuç açısından en iyi yöntemlerden biridir.

3. GENETİK ALGORİTMALAR

3.1. Genetik Algoritma Kavramı

Algoritma, belirli tek bir problemi çözecek davranışın, varolan veya sonradan tanımlanan veri modeline dayandırılarak adım adım ortaya konulması ve bunun bilgisayar ortamında herhangi bir programlama diliyle kodlanmasıdır (Çölesen, 2002).

Günümüzde çözülmesi zor ya da olanaksız olan çok sayıda problem vardır. Bu problemlerin çözülmesinde geliştirilmiş bir matematiksel fonksiyon olmadığı gibi olası çözümlerin hesaplanması ve en iyi çözümün seçilmesi de çok zaman almaktadır. Olası tüm çözümlerin değil de salt bazı seçilmiş çözümlerin denenmesi yöntemiyle, beklenen optimum sonucun elde edilmesini sağlayacak algoritmalar araştırılmış ve pek çok yöntem geliştirilmiştir. Geliştirilen yöntemlerden biri de Genetik Algoritmalarıdır.

Genetik (Kalıtsal) Algoritma (GA), rassal arama tekniklerini kullanarak çözüm bulmaya çalışan, parametre kodlama esasına dayanan bir arama tekniğidir ve bir veri grubu içinde özel bir veriyi bulmak için kullanılır (Goldberg, 1989).

GA'lar doğada geçerli olan en iyinin yaşaması kuralına dayanarak sürekli iyileşen çözümler üretir. Bunun için "iyi"nin ne olduğunu belirleyen bir uygunluk fonksiyonu ve yeni çözümler üretmek için yeniden kopyalama, değiştirme gibi operatörleri kullanır. GA'ların bir diğer önemli özelliği de bir grup çözümle uğraşmasıdır. Bu sayede çok sayıda çözümün içinden iyileri seçilip kötülerini elenebilir.

3.2. Genetik Algoritmaların Çalışma İlkesi

Genel anlamıyla GA, bir araştırma konusu olup model hâline getirilmiş neden-sonuç işleminin tersine rastgele örnekleme olgusu altında modellenmiştir. Kontrol edilerek onaylanan bilgi, organizma olarak adlandırılan aday çözümler içerisinde saklanmıştır. Organizmalar popülasyon olarak adlandırılıp grup hâlinde yer almışlardır.

GA'lar ile ilgili temel kavramlar aşağıda kısaca anlatılmaktadır:

Gen: Kalıtsal molekülde bulunan ve organizmanın karakterlerinin tayininde rol oynayan kalıtsal birimlere denir. Yapay sistemlerde gen, kendi başına anlamlı bilgi taşıyan en küçük birim olarak alınır. Bu bildirinin çalışma alanı olan Sipariş Tipi Atölyelerde İş Sıralamada her bir gen makinalarda işlenecek iş sıralarını temsil eder.

Kromozom (Birey): Birden fazla genin biraraya gelerek oluşturduğu diziye denir. Kromozomlar, alternatif aday çözümleri gösterirler.

Popülasyon: Kromozomlardan oluşan topluluğa denir. Popülasyon, geçerli alternatif çözüm kümesidir. Popülasyondaki birey sayısı (kromozom) genelde sabit tutulur. GA'da popülasyondaki birey sayısı ile ilgili genel bir kural yoktur. Popülasyondaki kromozom sayısı arttıkça çözüme ulaşma süresi (iterasyon sayısı) azalır.

Bir problemin GA ile çözümünde izlenecek işlem adımları aşağıda verilmektedir (Croce ve diğ., 1995):

1. Kullanıcının önceden tanımladığı kurallara göre genellikle rassal bir çözüm grubu seçilir veya kullanıcı kendisi ilk çözüm grubunu belirleyebilir. İlk çözüm grubuna başlangıç popülasyonu denir.

2. Her bir kromozom için bir uygunluk değeri hesaplanır; bulunan uygunluk değerleri dizilerin çözüm kalitesini gösterir. Popülasyonda yer alan en iyi uygunluk değerine sahip olan birey (kromozom), bir sonraki yeni nesile (popülasyon) doğrudan değiştirilmeden aktarılır. Buna elitizm denir.

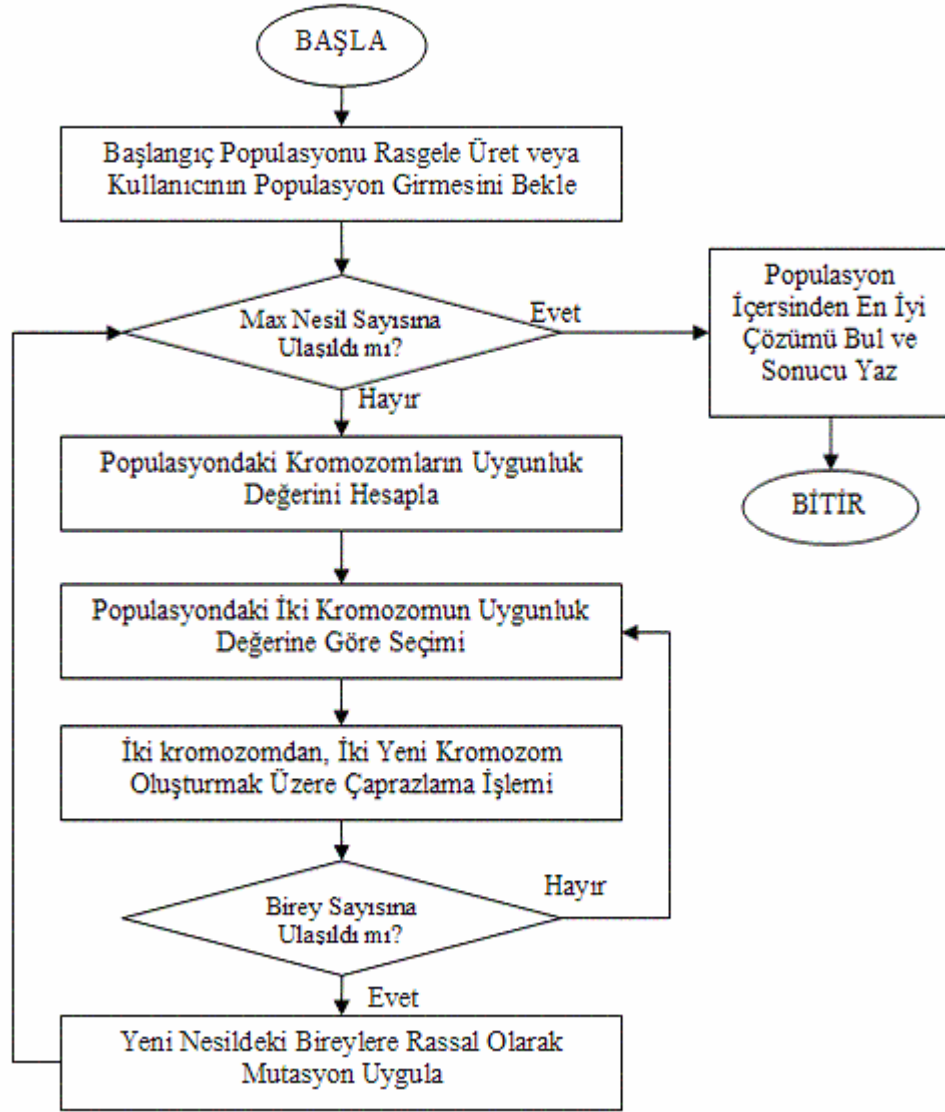
3. İki grup dizi (kromozom), belirli bir seçim yöntemine göre (uygunluk değerlerine göre hesaplanmış olasılık değerlerine göre) rassal olarak seçilirler.

4. Seçilen iki kromozom için rassal olarak genetik operatörler kullanılarak çaprazlama işlemi gerçekleştirilir. Sonuçta yeni popülasyonda yer alacak iki yeni birey (kromozom) oluşur. Çaprazlama, yeni popülasyonda yer alacak birey sayısına ulaşılan dek sürer.

5. Yeni popülasyondaki bireyler, rassal olarak mutasyon işleminden geçerler.

6. Önceden belirlenen nesil sayısı boyunca yukarıdaki işlemler sürdürülür. Eğer en büyük nesil sayısına ulaşılmamışsa adım-2'ye dönlür. İterasyon, en büyük nesil sayısına ulaşıncaya kadar işlem bitirilir. Uygunluk değeri en yüksek olan kromozom (çözüm) seçilir.

GA'lar, popülasyon adını verdiğimiz bir çözüm kümesi (kromozom grubu) ile başlar. Bir popülasyondan alınan çözümler, diğer bir popülasyonun oluşturulmasında kullanılırlar. Bu durumda yeni popülasyonun bir öncekinden daha iyi (bulmaya çalıştığımız en uygun sonuca daha yakın) olması umulur. Yeni çözümler oluşturmak için diğer popülasyondan alınan çözümler, problemin uygunluk ölçütüne göre belirlenir. Buradaki uygunluk, en uygun çözüme ne kadar yakın bulunduğu şeklinde düşünülebilir. Bu işlem belli bir koşul (en iyi çözümün bulunması) sağlanana dek sürer. Şekil 1.'de bu süreç gösterilmektedir.



Şekil 1. Genetik Algoritma akış diyagramı

GA bireylerin uygunluk ve iyiliklerine göre ayrılıp fark edilmesine gerek duyar. Bu sâyede üstün ve başarılı olanlar bir sonraki neslin bireylerini oluşturur. Çaprazlama, iki kromozomun biraraya gelerek genetik bilgi değişimi yapmasıdır. Mutasyon ise bir kromozomun taşıdığı genetik bilgide bir nedene bağlı olmaksızın değişme olmasıdır. Mutasyon aramada kısır döngüye girilmemesini sağlamak, toplulukta çözümü olmayan birbirine benzer bireylerden kurtulmak ve yeni alt optimal çözümler bulunmasını sağlamak için kullanılır (Holland, 1975).

3.3. Genetik Algoritmaların Parametreleri

3.3.1. Kodlama

GA bir probleme uygulanmadan önce verinin nasıl kodlanacağını ve olanaklı çözümler kümesinden örnek çözümlerin bilinmesi gerekir. Kodlama plânı GA'nın önemli bir kısmını oluşturur. Çünkü bu plân, bilginin çerçevesini şiddetle sınırlayabilir. Öyle ki probleme özgü bilginin bir kromozomsal gösterimiyle temsili sağlanır.

Kuracağımız genetik modelin hızlı ve güvenilir çalışması için kromozomların kodlanmasının doğru yapılması gerekmektedir. GA'ları uygulayacağımız sistemle ilgili uygun kodlama türünün seçilmesi gerekmektedir. GA'larda kullanılan kodlama tipleri şunlardır:

İkili Kodlama: Her kromozom bir ikili string'e sahiptir (0 ve 1). Bu string'deki her bit, çözümün belli karakteristiğini temsil eder veya tüm string bir sayıyı temsil eder.

Permütasyon Kodlama: Düzenleme problemlerinde kullanılır. Burada her kromozom, sayıları bir sırada temsil eden bir sayılar string'idir. Permütasyon kodlama, gezgin satıcı ve çizelgeleme problemleri için kullanışlıdır (Tablo 1.).

Tablo 1. Permütasyon kodlamalı kromozom örnekleri

Kromozom A	1 5 3 2 6 4 7 9 8
Kromozom B	8 5 6 7 2 3 1 4 9

Değer Kodlama: Gerçek sayılar gibi karmaşık değerlerin kullanıldığı problemlerde, ikili kodlama zor olduğu için doğrudan değer kodlanması kullanılabilir (Tablo 2.).

Tablo 2. Değer kodlamalı kromozom örnekleri

Kromozom A	1.2324 5.3243 0.4556 2.3293
Kromozom B	(back), (back), (right), (forward), (left)

Ağaç Kodlama: Bu yöntem gelişen, değişen programlar veya ifadeler için kullanılır. Örneğin GA. Ağaç kodlamada her kromozom, bazı nesnelere (örneğin fonksiyonlar ya da programlama dilindeki komutlar gibi) ağacıdır.

3.3.2. Seçim Yöntemi

Yeni popülasyonun oluşturulmasında kaç ferden seçileceği ve hangi fertlerin eşleşme için seçileceği, seçim fonksiyonuyla sağlanır. Seçim operatörü, yapay bir seleksiyondur (Goldberg, 1989). Seçim yöntemi, algoritmanın başarısı açısından önemlidir. Ebeveynler, uygunluk değerlerine göre eşleşmek üzere seçilirler. Bunun için kullanılan yöntemler aşağıdaki gibidir:

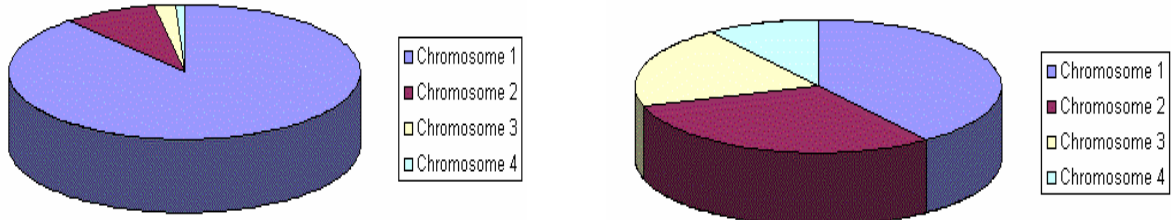
- Rulet-çemberi seçim yöntemi
- Rank Seçim Yöntemi
- Kararlı Hâl Seçim Yöntemi

Seçim yönteminde genellikle rulet çemberi yöntemi kullanılır. Varolan popülasyondaki en uygun kromozomlar rulet çemberi üzerine yerleştirilir, çember popülasyondaki dizi sayısı kadar döndürülerek çaprazlanacak bireyler elde edilir.

Şimdi bu yöntemler hakkında kısa açıklamalar yapalım:

Rulet-Çemberi Seçim Tekniği: Seçim yöntemleri içinde en basit ve en kolay uygulanabilir nitelikteki yöntem, rulet-çemberi yöntemidir. Rulet-çemberi yöntemi stokastik bir yöntemdir ve şu tekniği içermektedir: Tüm fertler bir çizgi üzerinde birbirine bitişik bölümler şeklinde dizilirler. Bununla birlikte her bir ferde ilişkin bölümün uzunluğu, onun uygunluk değeri kadar olur. Rastgele sayı üretilir ve rastgele sayı hangi bölüm içine denk gelirse, o bölümün ait olduğu fert seçilir. İşlem eşleşecek popülasyonun gerekli adedine ulaşıncaya dek sürdürülür. Bu teknik, üzerinde bölümleri olan rulet-çemberine benzediği için bu şekilde adlandırılmıştır. Bu teknikte, ebeveynler uygunluk değerlerine göre seçilirler. Kromozomlar ne kadar iyiye, seçilme şansları o kadar yüksektir. Bir kromozom birden fazla seçilebilir.

Rank Seçim Yöntemi: Yukarıdaki seçim eğer uygunluklar çok fazla değişiyorsa bazı sorunlara yol açacaktır. Örneğin en iyi kromozomun uygunluğu, tüm rulet tekerleğinin %90'ı ise diğer kromozomların seçilme şansları çok az olacaktır. Rank seçimi önce popülasyonu sıralar ve daha sonra her kromozom, uygunluğu bu sıralamadan sonra alır. En kötüsü 1 uygunluğunu alacak, ikinci en kötü 2 ve en iyisi N uygunluk değerini alacaktır. N de popülasyondaki kromozom sayısıdır. Sayıları düzenlemek için uygunlukları değiştirdikten sonra durumun nasıl değiştiğini Şekil 2.'deki gibi görebiliriz.



Şekil 2. Rulet Çemberi Yöntemi ve sonraki durum (Rank Seçim Yöntemi)

Bu yöntemde her kromozomun seçilme hakkı olacaktır. Ama bu yöntem daha yavaş çalışır, çünkü en iyi kromozomlar diğerlerinden fazla değişiklik göstermez.

Kararlı Hâl Seçim Yöntemi: “Yerine Geçme Yöntemleri” olarak da adlandırılabilirler. Bu, ebeveynleri seçmek için kısmî bir yöntem değildir. Bu seçimin ana düşüncesi, kromozomların büyük kısmının bir sonraki

nesilde hayatta kalmak zorunda olmasıdır. O zaman GA şu şekilde çalışır: Yeni çocuklar oluşturmak için her nesilde güzel ve iyi uygunluklu birkaç kromozom seçilir. Sonra kötü ve düşük uygunluklu bazı kromozomlar atılır ve yeni çocuk onun yerine yerleştirilir. Popülasyonun geri kalan kısmı yeni nesilde hayattadır. Yani kısaca bu yöntemde alt popülasyon oluşturulduktan sonra uygunluklar hesaplanır, en kötü kromozomlar yerlerini başlangıç popülasyonundaki en iyi kromozomlara bırakır (Engin, 2001).

3.3.3. Çaprazlama Oranı ve Çaprazlama Yöntemi

Çaprazlama oranı, fertlerin eşleştiklerinde çaprazlama yapıp yapmayacaklarına ilişkin olasılığı ifade eden orandır. Eğer eşleşme sonucunda çaprazlama da oluşursa, yeni ve genellikle ebeveynlerinden farklı bireyler elde edilmiş olur. Eğer çaprazlama gerçekleşmezse, bu takdirde ebeveynlerinin kopyası olan yeni fertler oluşur.

Çaprazlama sözcüğü, iki kromozomun (çözümün) birbirleri arasında gen alışverişinde bulunup iki yeni kromozom oluşturmasıdır. İkili yöntemle kodlanmış değişkenlerin yaptıkları üreme faaliyeti, kromozomların çaprazlamasına benzemesi dolayısıyla böyle adlandırılmaktadır. Ancak, gerçek değerlerin kullanılmak zorunda oldukları spesifik problemlerde, klâsik çaprazlama yöntemi yerine daha farklı yöntemler kullanılmaktadır. Seçim yöntemi ile yapay seçim sonucunda elde edilen yeni popülasyon dizisinden rassal olarak iki kromozom seçilir ve karşılıklı çaprazlama işlemine tâbi tutulur (Holland, 1975).

Örnek 1

Başlangıç popülasyonunda A_1 ve A_2 gibi iki dizi rassal olarak seçilmiş olsun.

$$\begin{aligned} A_1 &= 0\ 1\ 1\ 0\ 1 \\ A_2 &= 1\ 1\ 0\ 0\ 0 \end{aligned}$$

Bu örnekte dizi uzunluğu ($L=$) 5'dir. 1-5 arasında ($k=4$) seçildiğini varsayarsak, her iki dizinin de 4. genden sonra karşılıklı çaprazlama işlemine tâbi tutulacağı kabul edilir. Bu durumda yeni kromozomlar aşağıdaki gibi olur.

$$\begin{aligned} A_1' &= 0\ 1\ 1\ 0\ 0 \\ A_2' &= 1\ 1\ 0\ 0\ 1 \end{aligned}$$

En basit çaprazlama yöntemi, yukarıdaki örnekte verildiği gibi bir noktalı çaprazlama yöntemidir. Bir noktalı çaprazlama yapılabilmesi için her iki kromozomun da aynı gen uzunluğuna sahip olması gerekir. İki noktalı çaprazlamada ise kromozom iki noktadan kesilir ve karşılıklı olarak pozisyonlar yer değiştirir.

3.3.4. Mutasyon Oranı ve Mutasyon Yöntemi

Mutasyon oranı, tıpkı çaprazlama oranında olduğu gibi mutasyonun olasılığını ifade eden bir orandır. Yine rastgele yöntemlerle kromozomun mutasyona uğrayıp uğramayacağı belirlenir ve buna göre mutasyon gerçekleştirilir. Mutasyon oranı genellikle çok düşük (0,01) olduğundan mutasyon işlemi fertlerde az görülür.

GA'da önemli rol oynayan süreçlerden biri de mutasyon operatörüdür. Yapay sistemlerde mutasyon işlemi, kromozomlarda anı olarak oluşan değişimlerdir. Mutasyon sırasında kromozomdaki gen sayısı değişmez, sabit kalır. Mutasyon işlemi bir tek kromozom üzerinde yapılır. Mutasyon oranına göre, mutasyona uğratılacak sayıdaki diziler popülasyondan rassal olarak seçilir ve belirlenen mutasyon yöntemine göre değişime uğratılır.

3.3.7. Uygunluk Fonksiyonu

Uygunluk fonksiyonu GA'nın yapısında en temel yeri oluşturmaktadır. Bu fonksiyon ile kromozom içindeki bilgiler çözülerek sayısal bir değer elde edilir, bir başka deyişle kromozomun sisteme uygulanması durumunda sistemin vereceği çıktı bulunur. Uygunluk fonksiyonu, GA uygulamalarında yön tayin edici, stratejik ve çok önemli bir pozisyonda bulunduğundan, belirlenmesi uzun emek ve uzun zihinsel çaba gerektirmektedir.

3.4. Genetik Algoritmaların Uygulama Alanları

GA'ların en uygun olduğu problemler, geleneksel yöntemler ile çözümü olanaklı olmayan ya da çözüm süresi problemin büyüklüğü ile üstel orantılı olarak artanlardır. Bugüne kadar GA ile çözümüne çalışılan konulardan bazıları şunlardır: Optimizasyon, biyo-bilişim, hüresel programlama , oyun programcılığı, plânlama problemleri, yönetim problemleri, doğrusal olmayan problemler, zamanlama problemleri, otomatik programlama, makina öğrenmesi, ekonomi, popülasyon genetiği, evrim ve öğrenme, sosyal sistemler, müzik.

3.5. Genetik Algoritmaların Diğer Yöntemlerle Karşılaştırılması

GA'lar, "Yapay Zekâ"nın gittikçe genişleyen bir kolu olan evrimsel hesaplama tekniğinin bir parçasını oluşturmaktadır. GA geleneksel yöntemlerle çözümü zor veya olanaksız olan problemlerin çözümünde kullanılmaktadır. GA'lar, çokça işlem gücü gerektirmektedir. Ancak problemin nasıl çözüleceğinin ya da nasıl hızlıca çözüleceğinin bilinmesini gerektirmeme avantajını da kullanıcıya sunmaktadır (Goldberg, 1989). GA'lar ile klâsik analitik yöntemler arasındaki fark, Tablo 3.'de gösterilmiştir.

Tablo 3. Genetik Algoritma Yaklaşımı ile analitik yöntemin karşılaştırılması

Özellik	Analitik-Spesifik Yöntem	Genetik Yaklaşım
Hız	Çözüme bağlı, genelde iyi	Orta, çözüme yakın
Performans	Çözüme bağlı	Neredeyse mükemmel
Problemi Bilme	Gerekir	Gerekmez
Gerekli İnsan Emeği	Birkaç dakikadan başlar, uzayabilir	Birkaç gün
Uygulanabilirlik	Düşük (Çoğu problemin kullanılabilir matematiksel parametreleri olmaz ya da hesaplanamaz niteliktedir)	Genel Uygulanabilirlik
Adımlar	Çözüm Değildir (hesaplama işlemi bitene kadar beklenebilmelidir)	Çözümdür (hesaplama işlemi istenilen zamanda kesilebilir, en son kalanlar iyi çözümdürler ve bunların en iyisi, bulunan en iyi çözümdür)

Günümüzde rastgele aramaların kullanımı artmaktadır. Bu tip aramalar eniyilemenin daha iyi yapılması amacını sağlamakta daha başarılıdır. İnsanların bilgisayarlardan genel beklentisi mükemmellik olduğu için bu tip aramalar başarısız görülebilir. GA'lar klâsik yöntemlerin çok uzun zamanda yapacakları işlemleri kısa bir zamanda çok net olmasa da yeterli bir doğrulukla yapabilir. Bir GA programını birkaç kez çalıştırarak (bir önceki bulunan sonucu da popülasyona ekleyerek) en uygun çözümü bulmak olanaklı olabilmektedir.

4. GENETİK ALGORİTMALAR KULLANILARAK SİPÂRİŞ TİPİ ATÖLYELERDE İŞ SIRALAMA YAPILMASI

4.1. Genel Bilgi

Sipârîş Tipi Atölyelerde İş Sıralama problemlerinde genel amaç permütasyon tipi her bir işin her bir makinadaki yapılış sırasını belirleyerek en kısa tamamlanma süresini elde etmektir.

4.2. Parametre Kodlama

Son on yıl boyunca değişik kodlama teknikleri geliştirilmiştir. Çözümün kromozomlara nasıl kodlanacağı GA'da anahtar niteliğinde bir çalışmadır. Kromozomların kodlanması ve şifre çözümü aşamasında, üç kritik noktaya dikkat edilmesi gerekir (Cheng ve diğ., 1996):

1. Kromozomun uygunluğu kontrol edilir.
2. Kromozomun belirlenen koşulları (kısıtları) sağlayıp sağlamadığı kontrol edilir.
3. Kodlama haritasında, kromozomun tek olduğu kontrol edilir.

Koşulları sağlamayan bir kromozom, çözüm uzayındaki değerleri şifreleyemez; böyle bir kromozom, evrime uğrayamaz. GA ile kurulacak modelde, iş çizelgeleme problemi ile ilgilenileceği için aşağıda bu konuda geliştirilen kodlama yöntemleri anlatılmaktadır. İş Sıralama problemleri ile ilgili olarak 9 tip temsil yöntemi geliştirilmiştir:

1. Operasyona dayalı kodlama
2. İşe dayalı kodlama
3. Liste önceliğine dayalı kodlama
4. İş çifti ilişkisine dayalı kodlama

5. Öncelik kuralına dayalı kodlama
6. Tamamlanma zamanına dayalı kodlama
7. Makinaya dayalı kodlama
8. Rassal anahtarlı kodlama
9. Karışık kodlama

Uygulamamızda (programda) Operasyona Dayalı Kodlama yöntemi kullanılmaktadır. Bu kodlama ilk olarak Davis tarafından çizelgeleme problemlerinde kullanılmıştır. Çizelgeleme problemi, n iş- m makina için, m alt kromozomdan oluşturulur ve her bir alt kromozom dizisindeki semboller n iş sayısı uzunluğunda olur. Her bir sembol, makinadaki ilgili bir operasyonu gösterir. Makina önüne, kuyruktaki bekleme durumunu analiz etmek için, simülasyonla kromozomlardan asıl çizelge çıkarılır. Bir kromozomun, [(2 3 1) (1 3 2) (2 1 3)] olduğu varsayılır ise, m_1 makinası için ilk gen (2 3 1) işleri tercih listesidir; m_2 makinası için (1 3 2) işleri ve m_3 makinası için ise (2 1 3) işleridir (Davis, 1985).

4.3. Seçim Operatörü

GA'nın yürütülmesinde düşünülecek diğer önemli bir etmen, gen havuzunda yer alacak kromozomların seçim olasılığıdır. Bu olasılıkları belirlemede GA modelinde kullanılan üreme (seçim) operatörleri şunlardır:

- Makina Verimlerine Bağlı Rulet Çemberi Yöntemi
- Toplam Akış Süresine (makespan) Göre Rulet Çemberi Yöntemi
- Ters Yapay Seçim Yöntemi

Uygulamamızda kromozom seçiminde Toplam Akış Süresine Göre Rulet Çemberi yöntemi kullanılmaktadır. Bu yöntemde, uygunluk fonksiyonunun kromozomu çözerek hesapladığı toplam akış süresi kullanılır. Popülasyondaki her bireyin seçim olasılığı aşağıdaki yordama göre hesaplanır (Gen ve Cheng, 2000):

1. Popülasyondaki her birey için uygunluk fonksiyonu ile toplam akış süresi hesaplanır.
2. Popülasyondaki toplam en büyük akış süresi (F_{enb}) bulunur.
3. Her bireyin toplam akış süresi ve F_{enb} arasındaki farka göre seçim olasılığı hesaplanır. Bir popülasyondaki tüm bireylerin seçim olasılığının toplamı 1'dir. Her bireye ilişkin olmak üzere $[0, 1]$ arasında seçim aralıkları oluşturulur.
4. Bilgisayara rastgele bir sayı ürettirilir ve rastgele sayıya karşılık gelen birey seçilir.

4.4. Çaprazlama Operatörü

Çaprazlama operatörü, GA'nın temel işlemcisi olup, GA'nın performansını büyük ölçüde etkiler. Akış tipi çizelgelemede genellikle bir ve iki noktalı çaprazlama yöntemi kullanılır. Akış Tipi Çizelgelemede kullanılan çaprazlama operatörleri şunlardır:

- Pozisyona Dayalı Çaprazlama
- Sıraya Dayalı Çaprazlama
- Kısmî Plânlı Çaprazlama (PMX)
- Dairesel Çaprazlama (CX)
- Doğrusal Sıralı Çaprazlama (LOX)
- Alt Değişimli Çaprazlama (SXX)
- Sıralı Çaprazlama
- İş Tabanlı Sıralı Çaprazlama

Uygulamamızda çaprazlama operatörü olarak tek ve çift noktalı çaprazlama yöntemi ve en etkili yöntemlerden biri olan Alt Değişimli Çaprazlama tekniği kullanılmaktadır. Alt Değişimli Çaprazlama'da, bir iş sırası matrisinde, n iş- m makina probleminde $n*m$ matris kodu yazılabilir. Bu koddaki her bir gen, bir iş operasyonuna karşılık gelmektedir. Operasyon şeklinde kodlanan çizelgeleme problemlerinde iki işlem adımı ile SXX çaprazlama yöntemi kullanılır (Cheng ve diğ., 1996).

SXX çaprazlama yöntemini bir örnekle açıklayalım. Çaprazlama yapılacak iki kromozomumuz aşağıdaki gibi olsun:

A : 7 4 1 8 5 2 9 6 3 B : 3 5 7 6 8 2 4 9 1

Rastgele olarak seçilen işlerimiz (allel'lerimiz): 1, 3 ve 5 olsun. A kromozomunda bu işler (1 5 3) sırasıyla, B kromozomunda ise (3 5 1) sırasıyla bulunmaktadır. Buna göre alt değişimli çaprazlama sonucunda şu bireyleri elde ederiz:

$A^1 = 7 4 3 8 5 2 9 6 1$ $B^1 = 1 5 7 6 8 2 4 9 3$

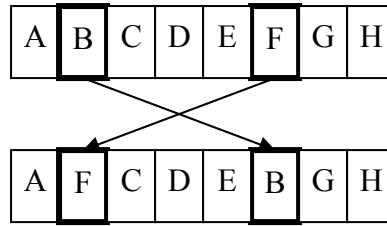
4.5. Mutasyon Operatörü

GA'larda önemli rol oynayan etmenlerden biri de mutasyon operatörüdür. Mutasyon sıklığının belirlenmesini sağlayan Mutasyon Oranı, doğal popülasyonlarda mutasyon oranı çok düşük olduğundan, GA'da da genelde düşük seçilir.

Son on yılda çok çeşitli mutasyon yöntemleri geliştirilmiştir. İş çizelgeleme ile ilgili geliştirilen yöntemler şunlardır (Wang ve diğ., 1999):

- Ters Mutasyon
- Komşu İki İşi Değiştirme
- Keyfi İki İşi Değiştirme
- Keyfi Üç İşi Yer Değiştirme
- Araya İş Ekleyerek (Shift) Değiştirme

Uygulamamızda kromozomların mutasyonu için Ters Mutasyon yöntemi kullanılmaktadır. Bu yöntemde, bir kromozomda rassal olarak iki pozisyon seçilir, bu iki pozisyonadaki alt diziler ters çevrilir. Şekil 3.'de buna bir örnek görülmektedir.



Şekil 3. Ters mutasyon

5. HAZIRLANAN GENETİK ALGORİTMA YAZILIMI

5.1. Programla İlgili Temel Bilgiler

Visual Basic.Net 2002 ile yazılan programda amaç, Sipariş Tipi Atölyelerde İş Sıralama problemlerinde yayılma süresini enküçükleyen sonucu GA yöntemini kullanarak bulmaya çalışmaktır. Özetle, Sipariş Tipi Atölyelerde İş Sıralama problemlerinde makinaların operasyonları belli bir sıraya göre yapmasını bir sistem olarak görürsek, amaç, sistemin çıktısını yâni tüm işlerin bitirildiği zamanı enküçüklemektir. Tüm işlerin bitirildiği zaman ise yapılan son operasyonun bitiş zamanıdır.

Programda yer alan bazı GA ifadeleri ve gerçekte Sipariş Tipi Atölyelerde İş Sıralamadaki anlamları aşağıdaki gibidir:

Gen: Bir makinada işlenecek işlerin sırasıdır. Sipariş Tipi Atölyelerde İş Sıralamada genler permütasyon kodlama yanında liste önceliğine dayalı kodlama ile elde edilir. Örneğin, 5 iş-4 makinalı bir Sipariş Tipi Atölye İş Sıralama probleminde bir gen şu şekilde olabilir: 41523. Buna göre makina ilk önce 4 no.'lu işi, daha sonra 1 no.'lu işi, ... ve en sonra da 3 no.'lu işi işleyecektir. Program, iş sayısının 10'dan fazla olduğu problemlerde, genlerde 10 no.'lu iş yerine A, 11 no.'lu iş yerine B, ... karakterini kullanır. Bir örnek vermek gerekirse, 7A4B1852963 geni için makina ilk önce 7 no.'lu işi, ikinci olarak 10 no.'lu işi, üçüncü olarak 4 no.'lu işi ve dördüncü olarak 11 no.'lu işi yapacaktır.

Allel: Bir geni oluşturan birimdir. Örneğin 41523 geninde 5 ifadesi bir allel'dir. İş Sıralamada allel yerine "operasyon" ifadesi kullanılır. Bir gende iş sayısı kadar allel vardır.

Kromozom (Birey): Makinalarda işlenecek iş sıralarından (genlerden) oluşan üretim plânıdır. Örneğin 5 iş-4 makinalı bir Sipariş Tipi Atölyelerde İş Sıralama probleminde bir kromozom şu şekilde olabilir: 41523524133254124513. Bu kromozom şu anlama gelmektedir: İlk makinada işlenecek işlerin sırası 41523, ikinci makinada işlenecek işlerin sırası 52413, üçüncü makinada işlenecek işlerin sırası 32541 ve son makinada işlenecek işlerin sırası 24513'dür. Bir kromozomda makina sayısı kadar gen (iş sırası), "iş sayısı*makina sayısı" kadar da allel (operasyon) vardır.

Popülasyon büyüklüğü (Birey Sayısı): Bir nesilde denenecek kromozom sayısını verir.

Nesil Sayısı: Programın üreteceği son neslin numarasını temsil eder.

Çaprazlama: İki kromozom (üretim plânı) arasındaki gen (iş sırası) alışverişini temsil eder. Program tekli, çoklu ve alt değişimli çaprazlama olarak üç çeşit çaprazlama yapmaktadır. Program, kullanıcının girdiği "Alt Değişimli Çaprazlama Olasılığı"na bağlı olarak rastgele gerçekleştirilir. Örneğin kullanıcı "Alt Değişimli Çaprazlama Olasılığı"nu 0,5 girerse, tekli çaprazlama olasılığı $[(1-0,5)/2]=0,25$ ve çoklu çaprazlama olasılığı $[(1-0,5)/2]=0,25$ olacaktır. Sonuç da $[0,5+0,25+0,25]=1$ 'dir.

Çaprazlama Oranı: Seçilen iki kromozomun çaprazlamaya girip girmeyeceğinin rassal olarak belirlenmesini sağlayan 0-1 arası bir değerdir.

Mutasyon: Seçilen bir kromozomun (üretim plânı) herhangi bir geninde, rassal olarak seçilen iki allel arasındaki yer değişimine denir. Örneğin, 41523 geninde rassal olarak 1. ve 3.'nün yeri seçilmişse, mutasyon gerçekleştiğinde yeni gen 51423 olur.

Mutasyon Oranı: Kromozomun mutasyona girip girmeyeceğinin rassal olarak belirlenmesini sağlayan 0-1 arası bir değerdir.

Programda kullanılan GA yönteminin çalışma ilkesi şu şekildedir :

1. Başla: n adet kromozomdan (probleme uygun çözümler) oluşan bir popülasyon üret veya kullanıcının başlangıç popülasyonunu girmesini bekle.

2. Uygunluk: Her kromozomun uygunluğunu hesapla (her çözüme ilişkin yayılma süresini bul).

3. Yeni popülasyon üret: Aşağıdaki adımları yeni bir popülasyon oluşuncaya dek yinele.

a. Seçim: Herhangi bir popülasyondan uygunluk değerlerine göre iki ebeveyn seç (ne kadar az yayılma süresi, o kadar çok seçilme şansı).

b. Çaprazlama: Çaprazlama olasılığına göre bireyleri yeni birey oluşturmak için çaprazla. Eğer hiç çaprazlama olmazsa çocuk, ebeveyninin kopyası olacaktır.

c. Mutasyon: Mutasyon olasılığına göre yeni bireyi değişikliğe uğrat.

d. Onaylama: Oluşturulan bireyi yeni popülasyona yerleştir.

4. Popülasyonu değiştir: Algoritmanın bir sonraki adımında oluşturduğun yeni popülasyonu kullan.

5. Test et: Eğer bitiş koşulu sağlanmışsa (son nesile ulaşılmışsa) dur ve varolan popülasyon içindeki en iyi çözümü, çıktı olarak bildir.

6. Geri dön: Adım 2'ye dön.

5.2. Program Kod Çözücüsünün Çalışma Mantığı

Kod çözücü GA konusunda, kromozomdan aldığı kodları girdi olarak alıp, ona bağlı bir uygunluk değeri üreten bir alt sistem olarak düşünülebilir. Sipariş Tipi Atölyelerde İş Sıralamaya göre ise, operasyon sıraları bilinen plâna göre son işlenecek işin bitiş zamanını bulmaya yarayan bir algoritmadır. Kod çözücünün çalışma mantığını bir örnek üzerinde açıklayalım.

Örnek 2

Bir (n=3) iş ve (m=4) makina iş sıralama problemi için, süre ve sıralama bilgileri Tablo 4.'de ve Tablo 5.'de verilmektedir.

Tablo 4. Örnek 2.'ye ilişkin sıralama bilgileri

İş / Makina	Makina A	Makina B	Makina C	Makina D
İş X	1	2	3	4
İş Y	4	3	2	1
İş Z	2	1	4	3

Tablo 5. Örnek 2.'ye ilişkin süre (dk) bilgileri

İş / Makina	Makina A	Makina B	Makina C	Makina D
İş X	5	7	8	6
İş Y	4	5	4	3
İş Z	2	3	4	5

Programın oluşturduğu 132312213231 kromozomunu çözelim. Buna göre:

A makinasında iş sıraları : 132 → XZY

B makinasında iş sıraları : 312 → ZXY

C makinasında iş sıraları : 213 → YXZ

D makinasında iş sıraları : 231 → YZX

zaman(i) : i. makinada zamanın işleyiş durumu

issuresi(j, i) : j. işin, i. makinadaki işleme süresi

$$isbittimi(j, k) = \begin{cases} 1, & \text{eğer j. işin k. sırasındaki operasyonu yapıldı ise} \\ 0, & \text{aksi durumda} \end{cases}$$

Kod çözücümüz, şifreyi şu adımları izleyerek çözer:

1. Adım:

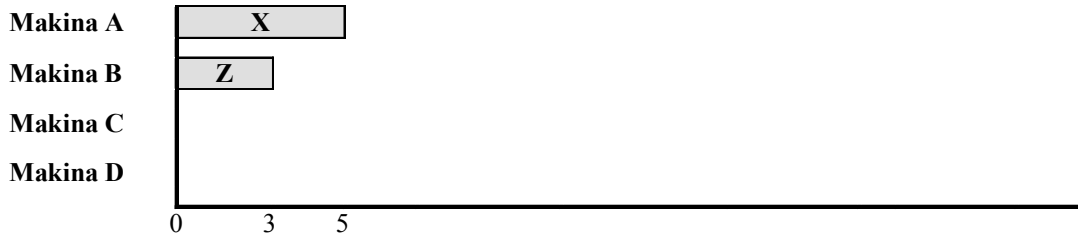
- I. İlk makinanın (A) (ilk genin) ilk allel'ini (operasyonunu) oku $\rightarrow X$
- II. X'in o makinada (A'da) hangi sırada olduğunu tablo yardımı ile bul $\rightarrow 1. sırada$
- III. İş makinada birinci sırada ise o makinada o işi başlat
- IV. zaman(i) (i. makinadaki zaman nerede işliyor?) değerini bul
 $zaman(A) = 0 + 5 = 5$ isbittimi (X, 1) = 1
- V. Bir sonraki makinaya geç (Şekil 4.)



Şekil 4. Birinci adıma ilişkin Gantt Şeması

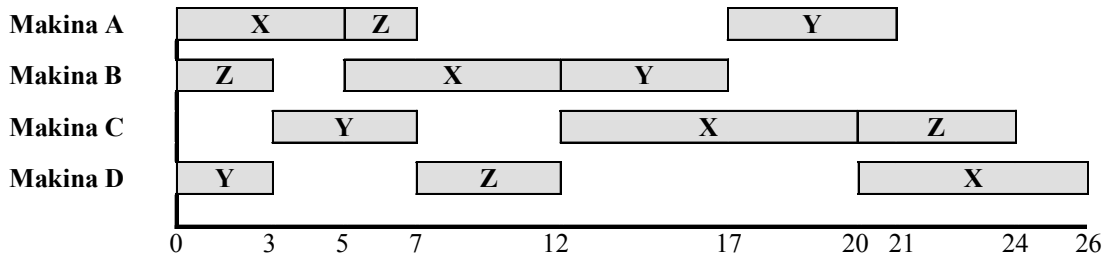
2. Adım:

- I. Sonraki makinanın (B) ilk allel'ini (operasyonunu) oku $\rightarrow Z$
- II. Z'nin o makinada (B'de) hangi sırada olduğunu tablo yardımı ile bul $\rightarrow 1. sırada$
- III. İş makinada birinci sırada ise o makinada o işi başlat
- IV. zaman(i) değerini bul
 $zaman(B) = 0 + 3 = 3$ isbittimi (Z, 1) = 1
- V. Bir sonraki makinaya geç (Şekil 5.)



Şekil 5. İkinci adıma ilişkin Gantt Şeması

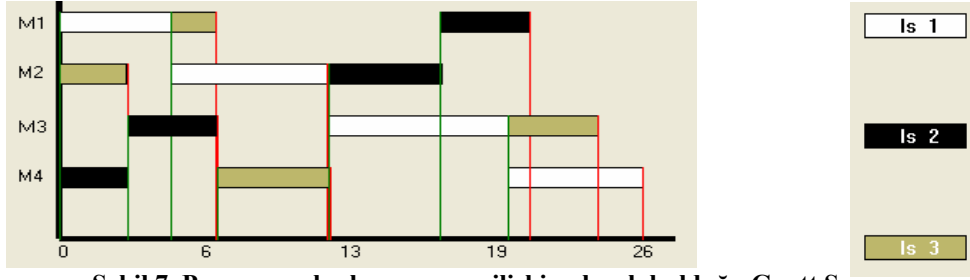
İşlemler bu şekilde sürdürülürse 14. adımın sonunda Şekil 6'daki durum elde edilir.



Şekil 6. Ondördüncü adıma ilişkin Gantt Şeması (Sonuç)

Uygunluk değeri (yayımla süresi) olarak bulunan tüm zaman(makina) değerlerinin en büyüğü olan [zaman(D)=] 26 seçilir.

İşlerin tüm operasyonları bittiği için kod çözücü, bir son operasyonun bitiş zamanı olan 26 sonucuna ulaşmıştır. Programın bu problem için çizdiği Gantt şeması ise Şekil 7.'de gösterilmektedir. İş 1, X'i; iş 2, Y'yi ve iş 3, ise Z'yi temsil etmektedir.



Şekil 7. Programın bu kromozoma ilişkin olarak bulunduğu Gantt Şeması

6. UYGULAMA

Arçelik Motor Fabrikasından alınan verilerle yapılan uygulamaya ilişkin bilgiler şu şekildedir: Atölyemizde 12 iş, 9 makina bulunmaktadır. Bu 12 işten, ilk 5 tanesi ağır (işlem süreleri uzun) olup, sonraki 7 iş ise daha hafiftir. Makinalar ise Tablo 6.'daki gibi gruplanmıştır. A grubu makinalardan Makina 1, diğer makinalara (Makina 2 ve 3) göre daha yeni olduğu için işleme süresi daha kısadır. Makina 4 ise aynı işi Makina 5'den daha kısa sürede işlemektedir. İşler genellikle A operasyonu ile başlamakta ve F operasyonu ile bitmektedir. İş sıraları Tablo 7.'de görüldüğü gibidir.

Tablo 6. Makinalar tablosu

A Grubu Makinalar	Makina 1, Makina-2, Makina-3
B Grubu Makinalar	Makina 4, Makina-5
C Makinası	Makina 6
D Makinası	Makina 7
E Makinası	Makina 8
F Makinası	Makina 9

Tablo 7. Makinalar bazında iş sıraları

İş No	A			B		C	D	E	F
	1. Makina	2. Makina	3. Makina	4. Makina	5. Makina	6. Makina	7. Makina	8. Makina	9. Makina
1	1	1	1	2	2	0	4	3	5
2	2	2	2	1	1	3	4	0	0
3	1	1	1	4	4	3	2	6	5
4	1	1	1	3	3	2	0	4	5
5	1	1	1	0	0	5	4	2	3
6	3	3	3	2	2	0	1	4	5
7	0	0	0	1	1	2	0	3	4
8	1	1	1	4	4	3	2	0	5
9	2	2	2	3	3	1	4	5	0
10	1	1	1	0	0	2	3	4	5
11	5	5	5	1	1	3	2	4	6
12	1	1	1	2	2	5	4	3	0

Tablo 7.'ye göre, ilk üç ve sonraki iki makina kendi aralarında aynı işlemi yaptığından sıralamada aynı numaralar ile gösterilmiştir. Bazı işler için ise bazı makinalarda işlemeye gerek görülmemektedir. Bu durum, sıralamada "0" ile gösterilmiştir. Örneğin 1. iş için operasyon sırası şu şekildedir:

1. A makinası : A grubu makinalardan herhangi biri
2. B Makinası : B grubu makinalardan herhangi biri
3. E Makinası
4. D Makinası
5. F Makinası

Görüldüğü gibi, 1. işin operasyonları arasında, 6. makina olan C makinası yoktur. Fabrikadan alınan işlem süresi (dk.) bilgileri Tablo 8.'de verilmiştir.

1435AB862C973475C9A186B253176C928AB47215A3C984B6B146258CA397976314AB825C63B1A
584C79251B8A742C9635C3AB1472869

Kromozoma göre makinalarda işlenecek iş sıraları, Tablo 10.'daki gibi bulunmuştur. Buna göre tüm işler 374 dakikada tamamlanacaktır.

Tablo 10. Çözüm tablosu

Makina Grubu	Makina No	İşler	Açıklama
A Grubu Makinalar	Makine 1	1 - 4 - 10 - 11 - 2	7. işin A operasyonu olmadığı için burada yer almıyor.
	Makine 2	3 - 12 - 9 - 8	
	Makine 3	5 - 6	
B Grubu Makinalar	Makine 4	7 - 2 - 3 - 12 - 9 - 8	5. ve 10. işlerin B operasyonu olmadığı için burada yer almıyor.
	Makine 5	11 - 1 - 4 - 6	
C Makinası	Makine 6	9 - 7 - 3 - 4 - 10 - 11 - 8 - 2 - 5 - 12	1. ve 6. işlerin C operasyonu yok.
D Makinası	Makine 7	6 - 3 - 11 - 1 - 10 - 5 - 8 - 12 - 9 - 2	4. ve 7. işlerin D operasyonu yok.
E Makinası	Makine 8	5 - 1 - 11 - 10 - 7 - 4 - 12 - 9 - 6 - 3	2. ve 8. işlerin E operasyonu yok.
F Makinası	Makine 9	5 - 3 - 10 - 11 - 1 - 4 - 7 - 8 - 6	2, 9. ve 12. işlerin F operasyonu yok

7. SONUÇ ve ÖNERİLER

Genel anlamda Üretim Plânlama, belirli miktar ve kalitedeki ürünlere olan istemi veya siparişleri karşılamak için varolan olanakları en uygun şekilde kullanan ve işletme amaçlarına olumlu yönde katkıda bulunan bir faaliyetler kümesidir. Genel olarak **Sipariş (Atölye) Tipi Üretim Sistemleri** ve **Akış (Ürün Hattı) Tipi Üretim Sistemleri** olmak üzere iki tip üretim organizasyonu vardır. Genellikle üretim sistemlerinde bu iki tipin karışımı olan bir örgüt yapısı görülmektedir. Tümüyle sipariş tipi veya akış tipi bir fabrika bulmak oldukça zordur. Bu nedenlerle en uygun çözümü bulunamayan akış tipi çizelgeleme problemlerinin GA ile çözüm performansının artırılması düşünülmüştür. Bu nedenle GA'larla Sipariş Tipi Atölyelerde İş Sıralama programı fabrikalarda üretim çizelgelerinin hazırlanmasında oldukça yararlı olacaktır. Ayrıca bu konu, Üretim Plânlamada yer alan en uyguna yakın iş çizelgelerinin hazırlanmasını amaçlayan Optimum Üretim Teknolojisi yaklaşımını da desteklemektedir.

Kaynaklar

Acar, N., *Üretim Planlaması Yöntem ve Uygulamaları*, Milli Prodüktivite Merkezi Yayınları No: 280, Ankara, 2001.

Cheng, R., Gen, M. Tsujimura, Y., A Tutorial Survey of Job Shop Scheduling Problems Using Genetic Algorithms-I, Representations, *Computers and Industrial Engineering*, Vol : 30, No: 4, p. 983 – 997, 1996.

Croce, F. D., Tadei, R., Volta, G., A Genetic Algorithm for Job Shop Scheduling, *Computers and Industrial Engineering*, Vol:25, No: 1-4, Pergamon, 1995.

Çölesen, R., Veri Yapıları ve Algoritmalar, *Papatya Yayıncılık*, İstanbul, 2002.

Davis, L., Job Shop Scheduling with Genetic Algorithm, In : *Proceeding of the First International Conference on Genetic Algorithms*, Hillsdale, NJ, Lawrence Erlbaum, p. 36-40, 1985.

Engin, O., Akış Tipi Çizelgeleme Problemlerinin Genetik Algoritmalar ile Çözüm Performansının Artırılmasında Parametre Optimizasyonu, *Doktora Tezi*, İTÜ Fen Bilimleri Enstitüsü, 2001.

Gen, M., Cheng, R., Genetic Algorithms and Engineering Optimization, *John Wiley Sons, Inc*, USA, 2000.

Goldberg, D. E., Genetic Algorithms in Search Optimization and Machine Learning, *Addison-Wesley Publishing Company*, USA, 1989.

Holland, J.H., Adaptation in Natural and Artificial Systems, *University of Michigan Press*, Ann Arbor, 1975.

Wang, D., Gen, M., Cheng, R., Scheduling Grouped Jobs on Single Machine with Genetic Algorithms, *Computers and Industrial Engineering*, 36, p. 309-324, 1999.