

SU KAYNAKLARI SİSTEMLERİNDE LİNEER OLMAYAN PROBLEMLERİN GENETİK ALGORİTMA İLE ÇÖZÜMÜ

Ahmet BAYLAR, Mualla ÖZTÜRK, Abdussamet ARSLAN
Fırat Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü, Elazığ

ÖZET

Genetik Algoritma tekniği, evrim sürecinden etkilenecek, canlılarda yaşanan genetik sürecin bilgisayar ortamında gerçekleştirilmesi işlemidir. İşlemler bilgisayar hafızasına depo edilmiş kromozomlar üzerinde icra edilmektedir. Çaprazlama operatörü vasıtasıyla, kromozomlar arasındaki genetik bilgi sürekli olarak değişmekte ve topluluğun başarısı artmaktadır. Bu çalışmanın asıl amacı, su kaynakları sistemlerindeki lineer olmayan programlama problemlerinin Genetik Algoritma ile hesaplanmasıdır. Bu maksatla, Genetik Algoritma esaslı bir optimizasyon programı geliştirilmiştir. Elde edilen sonuçlara göre Genetik arama metodunun hassas sonuçlar verdiği gözlenmiştir.

Anahtar Kelimeler : Genetik algoritma, Optimizasyon, Lineer olmayan programlama

SOLUTION OF NONLINEAR PROBLEMS IN WATER RESOURCES SYSTEMS BY GENETIC ALGORITHM

ABSTRACT

Genetic Algorithm methodology is a genetic process treated on computer which is considering evolution process in the nature. The genetic operations takes place within the chromosomes stored in computer memory. By means of various operators, the genetic knowledge in chromosomes change continuously and success of the community progressively increases as a result of these operations. The primary purpose of this study is calculation of nonlinear programming problems in water resources systems by Genetic Algorithm. For this purpose a Genetic Algorithm based optimization program were developed. It can be concluded that the results obtained from the genetic search based method give the precise results.

Key Words : Genetic algorithm, Optimisation, Nonlinear programming

1. GİRİŞ

İnsanoğlu tarih boyunca hep doğayı taklit ederek belli hedeflere ulaşmıştır. Son yıllarda insanoğlu tarafından yapılan bazı araçların neredeyse doğadaki benzerlerinin bire bir mekanik veya elektronik taklitleri şeklinde oldukları açıkça görülmektedir. İnsanın doğasında olan bu davranış şekli ile insan zekasının yoğunlaşması sonucu elde edilenler, insanoğlunu günümüzdeki teknoloji düzeyi ve modern toplum yapısına ulaştırmıştır.

Mühendislik problemleri genellikle amaca yönelik olarak en iyi çözümü arayan lineer, lineer olmayan ve dinamik programlama problemlerinden ibarettir.

Bu nedenle optimizasyon yöntemleri mühendislik açısından özel bir önem arz etmektedir. Bilgisayarların yaygınlaşmasından önce çok küçük optimizasyon problemlerinin çözümünde dahi büyük güçlüklerle karşılaşmış, elle çözüme uygun birçok optimizasyon yöntemi geliştirilmiştir. Bilgisayarların yaygınlaşması ile birlikte çoğu problemler daha kolay ve daha kesin çözümlere kavuşmuş, ancak büyük ve karmaşık problemlerin çözümünde bilgisayarlar da yetersiz kalmıştır. Yetersizliklerin temel nedenleri, problemlerin analitik çözümlerinin olmayışı ve bilgisayarlarla iterasyona dayalı çözüm yöntemlerinde ortaya çıkan hız sorunlarıdır. Ortaya çıkan sorunlar insanları yeni çözüm yöntemleri arayışlarına sevk etmiştir.

Son yıllarda bilgisayar donanım ve yazılımlarının gelişmesine paralel olarak, düşünebilen, öğrenebilen ve karar verme yeteneğine sahip bilgisayarların yapılması en çok özlenen hedeflerden biri haline gelmiştir. Kısmen gerçekleşen bu hedeflerin kilometre taşlarını yapay zeka konuları oluşturmaktadır. Gündemi yoğun olarak işgal eden yapay zeka konularının başlıcaları Genetik Algoritmalar, Yapay Sinir Ağları, Uzman Sistemler, Isı Yayılım Simülasyonu ve Nesneye Yönelik Programlama gibi konulardır. Bu konuların hemen hemen hepsindeki en belirgin özellik, insan, doğa ve sosyal çevre yapılarının taklit edilmesine dayanmalarıdır.

Genetik Algoritmalar, insan ve ekosistemdeki doğal gelişme, sosyal sistemleri taklit etme ve psikolojideki sonuçları değerlendirmeyi içine alan dinamik gelişmelerin modellenmesinden oluşmaktadır. Modellenen örneklerin bir zaman dilimi içerisinde kötüden iyiye doğru gitmesi, bir optimizasyon probleminin başlangıç çözümünden optimum çözüme doğru yaklaşımını andırır.

Su kaynakları mühendisliğinde genel olarak Yapay Zeka bilim dalının uygulamaları literatürde oldukça az yer almaktadır. Yapılan uygulamaların büyük bir çoğunluğunu Uzman Sistemler oluşturmaktadır. Yapay Zekanın bir alt dalı olan Genetik Algoritmalar konusunda yeterli sayılabilecek derecede uygulama mevcut değildir. Oysa su kaynakları mühendisliğinde diğer mühendislik dallarının çoğunda olduğu gibi gözönüne alınan sistemin matematik modeli kurularak sistemin performansı belirlenmekte ve sistem analizi yöntemlerini kullanarak optimizasyon yapılmaktadır. Birçok hallerde birden fazla amacı birlikte gözönüne almak gerektiğinden optimum çözüm belirlenememektedir, ancak her bir seçeneğin çeşitli amaçlar açısından değeri belirlenebilmektedir.

Bu çalışmada, su kaynakları sistemlerindeki lineer olmayan programlama problemlerinin çözümünde Genetik Algoritma yöntemi kullanılmıştır. Konunun iyi bir şekilde anlaşılabilmesi için sınırlayıcı ve sınırlayıcı lineer olmayan programlama problemlerinden seçilen iki basit örnek incelenmiştir.

2. GENETİK ALGORİTMA

Genetik Algoritmalar, insan ve ekosistemlerdeki doğal gelişme, sosyal sistemlerdeki taklit etme ve psikolojideki sonuçları değerlendirmeyi içine alan dinamik metodların geniş bir şekilde modellenmesinden oluşmaktadır. Evrim

sistemlerinin bilgisayarda modellenmesini yapmak çoğu konvansiyonel modellemelere göre biraz daha zor olmaktadır (Arslan ve ark., 1996).

Genetik Algoritma ilk ismini biyoloji, ikinci ismini ise bilgisayar biliminden almaktadır (Holland, 1975). Tek bir mekanik yapının öğrenme yeteneğinin geliştirilmesi yerine, böyle yapılardan oluşan bir topluluğun çoğalma, çiftleşme, değişim vb. genetik süreçlerden geçerek, başarılı (öğrenebilen) yeni bireylerin oluştuğu görülmüştür. Holland'ın çalışmalarının sonuçlarını açıkladığı kitabının 1975'de yayınlanmasından sonra geliştirdiği yöntemin adı Genetik Algoritmalar ya da kısaca G.A. olarak yerleşmiştir. Ancak David E. Goldberg adlı inşaat mühendisi 1989'da konusunda bir klasik sayılan kitabını yayınlıncaya kadar, Genetik Algoritmalar'ın pek pratik yararı olmayan bir araştırma konusu olduğu düşünülmekteydi. Halbuki Goldberg'in gaz borusu hatlarının denetimi üzerine yaptığı çalışma ona sadece 1985 National Science Foundation Genç Araştırmacı ödülünü kazandırmakla kalmayıp, Genetik Algoritmalar'ın pratik kullanımının da olabilirliğini kanıtlamıştır (Goldberg, 1983).

Genetik Algoritma, bilgisayar üzerinde oluşan bir evrim şeklidir. Genetik Algoritmanın amacı hem problemleri çözmek hem de evrimsel sistemleri modellemektir. Değişik planlama teknikleri, bir fonksiyonun optimizasyonu veya ardışık değerlerin tespitini içine alan bir çok problem tipleri için çözüm geliştirmektedir. Genetik Algoritma ile oluşturulan seleksiyon, doğal topluluklara benzer bir şekilde bilgisayar hafızasına depo edilmiş kromozomlar üzerinde icra edilmektedir. Bilgisayara uyarlama tabiatla karşılaştırılmayacak kadar basitleştirilmesine rağmen, Genetik Algoritmalar, karışık olmasının yanında hayret uyandıracak kadar da ilginç yapıya sahiptirler.

3. GENETİK ALGORİTMANIN TEMEL PRENSİPLERİ

Ünlü matematikçi J. Hadamard "Gerek matematikte, gerek başka alanlarda buluş ve icatlar farklı alanlardan düşüncelerin biraraya getirilmesiyle gerçekleşir" demiştir. Bu söz Genetik Algoritmalar'ın nasıl doğduğunu özetlediği gibi temel çalışma ilkesini de açıklamıştır. Şekil 1'de Genetik Algoritma'nın yapısı genel hatları ile verilmektedir. Şekil 1 aslında Genetik Algoritma'nın toplumsal bakış açısı ile verilmiş yapısını göstermektedir. Genel olarak bu, Genetik

Algoritmaları da içine alan evrimsel algoritma kavramıdır.

1. Başlangıç anını belirle
 $t := 0$;
2. Toplulukta yer alacak bireyleri oluştur.
 $P(t)$;
3. Bireylerin yaşam koşullarına uygunluğunu değerlendir.
Değerlendir $P(t)$;
4. İstenilen düzeye ulaşılan kadar
While not başarılı-birey-bulundu do
5. Zaman sayacını artır.
 $t := t + 1$;
6. Bir sonraki topluluğa döl verecek bireyleri seç ve geçici topluluğa kopyala
 $P'(t) := \text{Ebeveynleri seç } P(t)$;
7. Seçilen bireyleri çiftleştir.
Çiftleştir $P'(t)$;
8. Geçici toplulukta rastgele değişimler olmasını sağla
Değişime- uğrat $P'(t)$;
9. Geçici topluluğun başarı durumunu değerlendir. Değerlendir $P'(t)$;
10. Bir sonraki topluluğu oluştur.
 $P(t+1) := P'(t)$;
End do ;

Şekil 1. Genetik algoritma'nın toplumsal bakış açısından yapısı

Buradaki terimlerde topluluk havuz ile, birey kromozom ile, çiftleşme çaprazlama ile, uygunluk fonksiyonu tasarımın kalitesi ile, sosyal başarı da tasarımın optimalitesi ile yer değiştirdiğinde ortaya Genetik Algoritmalar çıkmaktadır. Şekilde görüldüğü gibi başlangıçta yapılan işlem topluluğu oluşturmaktır. Bu adımda genellikle uygulanan yöntem başlangıç topluluğunu rastgele oluşturmaktır. Fakat topluluk oluşturulurken bireylerin mümkün olduğu kadar birbirine benzememesini sağlamak faydalı olmaktadır.

Toplulukları evrim sürecine sokmadan önce yapılması gereken bir başka işte, başlangıç bireylerini değerlendirmektir. Bu evrim süreci içerisinde bir sonraki nesle döl verecek olanların belirlenmesi gerekmektedir. Daha sonra istenilen düzeyde başarılı birey bulununcaya kadar veya topluluk başarıda artış sağlayamaz duruma gelince veyahutta önceden belirlenen evrim sayısı tamamlanuncaya kadar sürecek olan evrim başlayacaktır. Her evrim sürecinde tekrar edilecek olan işlemler de aşağıdaki şekilde sıralanacaktır.

Bir sonraki nesle döl verecek olan bireyler daha önce hesaplanmış olan başarı değerlerine bağlı olarak seçilmektedir. Bir sonraki işlem olarak, seçilen bireyler istenilen bir yöntemle çiftleştirilmektedir. Çiftleştirme sonucu oluşturulan

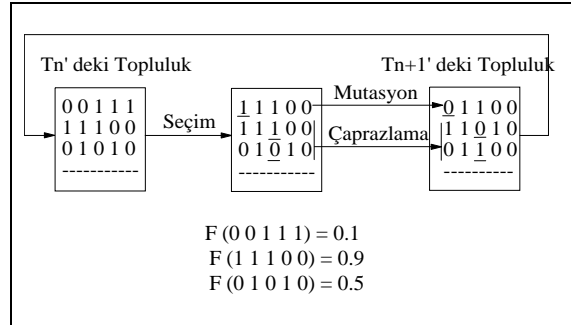
bireyler genellikle, % 0.1 olasılıkla değişime uğratılmaktadır. Son adım olarak da oluşturulan yeni bireylerin başarı değeri hesaplanmaktadır.

4. GENETİK ALGORİTMA OPERATÖRLERİ

Genetik Algoritma operatörleri; gen havuzu, seçim, çaprazlama ve mutasyondan meydana gelmektedir.

4. 1. Gen Havuzu

Gen havuzu, kromozomların uygunluklarının değerlendirilip kopyalama işleminin yapıldığı yerdir. Şekil 2'de gen havuzunda yapılan işlemler görülmektedir. Burada $F(00111) = 0.1$, $F(11100) = 0.9$, $F(01010) = 0.5$ topluluğun uygunluk değerlerini göstermektedir. Bu topluluk içerisinde uygunluk değeri yüksek olan bireyler seçilerek, en yüksek uygunluklu birey, uygunluğu düşük olan bireyin yerini almaktadır. Daha sonraki adımda ise ilk birey mutasyona uğratılmakta, ikinci ve üçüncü bireyler ise çaprazlama işlemine tabi tutulmaktadır. Bir sonraki adımda yeni bireyler oluşmaktadır. Bu işlemler daha önce verilmiş bir generasyon sayısına kadar veya topluluk artık başarıda artış sağlayamaz duruma gelinceye kadar devam etmektedir.



Şekil 2. Genetik algoritma işlemleri

Genetik Algoritma ile yapılan işlemler genellikle aşağıda belirtilen üç operatör yardımıyla icra edilmektedir.

4. 2. Seçim

Bu operatör yeni topluluk içerisinde, uygunluğu yüksek bireylerin bulunmasını sağlamaktadır. Burada bahsedilen uygunluk değeri, Genetik Algoritma ile bulunan problemin sonuç değerini göstermektedir. Bireylerin uygunluk değerleri esas alınarak seçim işlemi yapılmaktadır. Seçim işlemi topluluk içerisinde uygunlukları düşük olan bireyleri eleyip, elenenler yerine uygunlukları yüksek olan

bireylerin birkaç adet kopyasını yaparak tamamlanmaktadır.

4. 3. Çaprazlama

Biyolojik terim olarak çaprazlama genel olarak, bir bireyin içerisindeki değişimi kastetmesine rağmen, çaprazlama terimi burada bireyler arasındaki benzer alt kromozomların değişimlerini ifade etmektedir. Genetik Algoritmada çaprazlama iki kromozomun bir araya gelerek genetik bilgi değişimi yapmasıdır.

İki ebeveyn arasında seçilmiş olan sitelerdeki genler yerdeğiştirerek çaprazlama işlemi tamamlanmaktadır. En çok kullanılan çaprazlama çeşitleri bir noktalı çaprazlama, iki noktalı çaprazlama, üniform çaprazlama ve sıralı çaprazlamadır.

Yavru kromozomlar ebeveynlerden farklı olmasına rağmen onların özelliklerini taşımaktadır. Aşağıda görüldüğü gibi, kromozom uzunluğu $L = 10$ olan iki ebeveyn arasında 6. ve 10. sitelerde yapılan çaprazlama neticesinde iki yavru birey oluşmaktadır.

1. Ebeveyn = 1 1 0 0 1 0 0 0 0 0 1 1.Çocuk = 1 1 0 0 1 1 0 0 0 1
2. Ebeveyn = 0 1 0 1 1 1 0 0 0 1 2.Çocuk = 0 1 0 1 1 0 0 0 0 0

4. 4. Mutasyon

Sınırlı bir topluluk üzerinde çalışıldığında, toplulukta birkaç genetik bilginin erkenden kaybolma ihtimali bulunmaktadır. Örnek olarak, bir kromozomu oluşturan genlerin tamamı 0 yada 1 olabilmektedir. Böyle bir kromozomu çaprazlama operatörü ile değiştirmek mümkün olmamaktadır.

Çaprazlama vasıtasıyla üretilmeyen uygunluk değeri yüksek kromozomları, mutasyon vasıtasıyla üretmek mümkün olmaktadır. Bunun yanında uygunluk değeri oldukça yüksek olan kromozomları bozma ihtimali de bulunmaktadır. Aşağıda görüldüğü gibi $L = 8$ uzunluğundaki bir kromozomda, rastgele seçilmiş olan 5'nolu sitede, 1 değeri yerine 0 yazılarak kromozom mutasyona uğratılmıştır.

0 1 0 0 1 1 0 1 0 1 0 0 0 1 0 1

5. SU KAYNAKLARI SİSTEMLERİNİN OPTİMİZASYONUNDA GENETİK ALGORİTMA KULLANILMASI

Su kaynaklarını geliştirme çalışmalarında amaç fonksiyonunu en büyük yapacak çözümün (optimal proje büyüklüğü, proje bileşenlerinin optimal

boyutları, optimal işletme kuralları) çeşitli kısıtlara (teknolojik, ekonomik,...) uyarak belirlenmesi istenir.

Su kaynaklarını geliştirme çalışmalarında çok rastlandığı gibi optimize edilecek sistemin bileşenlerinin veya maksatlarının çok sayıda olması halinde üretim fonksiyonu üzerinde bulunan çeşitli seçenekleri karşılaştırarak bunların arasında en iyi çözümü bulmak için özel yöntemler kullanmak gerekir. Bu yöntemler şu şekilde sınıflandırılabilir :

5. 1. Klasik Yöntemler

Mühendislik çalışmalarında günümüzde çok sık kullanılan bu yöntemler deneyim, bilgi ve sezgiye dayanarak uzun yıllardır denenmiş ve alışılmış deneme yanılma teknikleri ile en iyi çözüme yaklaşmayı amaçlar. Genellikle basitleştirici kabuller yapılması gerektiği gibi varılan çözümün gerçekten optimum olduğu da garanti edilemez.

5. 2. Matematik Optimizasyon Yöntemleri

Nisbeten bazı basit problemlerin optimum çözümleri sistem analizinin lineer programlama, lineer olmayan programlama ve dinamik programlama gibi algoritmaları ile kesin olarak belirlenebilir. Ancak bu yöntemlerin su kaynakları sistemlerine uygularken yapılması gereken kabuller çoğu zaman sistemin gerçek niteliğinden ayrılmasına yol açtığından varılan çözümler de gerçekçi olmamaktadır. Hesap süresini gerçekçi ölçüler içinde tutabilmek için basitleştirici kabuller yapılması gerekir.

5. 3. Simülasyon (Benzetim)

Sistemin davranışı bilgisayarda benzeştirilerek çeşitli seçeneklerin verecekleri sonuçlar belirlenip karşılaştırılır. Bu yöntemde basitleştirici kabuller yapmadan çok karışık sistemleri incelemek mümkündür. Ancak optimal çözümü veren bir algoritma mevcut olmadığından deneme yoluyla varılan çözümün en iyi çözüm olduğu garanti edilememektedir. Optimum çözüme yaklaşabilmek için karar değişkenlerine çeşitli değer takımları vererek çok sayıda seçeneği incelemek gerekir. Matematik optimizasyon yöntemleri özellikle seçeneklerin ilk elemesini yaparken kullanılırlar. İlk elemeyen geçen seçenekler benzetim teknikleri ile daha ayrıntılı bir şekilde değerlendirilir.

Genetik Algoritmalar özellikle optimizasyon problemlerinin çözümünde geniş bir kullanım alanı bulmuştur. Klasik optimizasyon algoritmalarının yetersiz veya yavaş kaldığı birçok problemde sonuca

ulaşabilmek için kullanılacak en iyi araçlardan birisi, Genetik Algoritmalarıdır. Genetik Algoritmaların klasik optimizasyon yöntemlerine göre bazı önemli avantajları vardır. Bunlar aşağıdaki gibi sıralanabilir :

1. Genetik Algoritma açısından problemin lineerlik, non-lineerlik ve süreklilik gibi özelliklerinin hiç bir anlamı yoktur.
2. Optimize edilecek amaç fonksiyonunun süreksiz olması halinde, süreksizlik noktalarında fonksiyonunun türevi de olamayacağından, türev almaya dayalı optimizasyon yöntemleri uygulanamamaktadır. Oysa Genetik Algoritma optimizasyonunda, işlemlere fonksiyonların türevi girmediğinden, böyle bir sakınca söz konusu değildir.
3. Genetik Algoritma ile optimizasyon işlemlerinde, karmaşık matematiksel ifadeler yer almamaktadır.
4. Değişkenlerinin alabilecekleri değerler, sonlu bir tamsayı listesi veya yeterli hassasiyette sonlu ve ayrık bir değer uzayından alınabilir.
5. Genetik Algoritmalar ile optimizasyon algoritmasının en önemli avantajlarından biri; problem ile optimizasyon algoritması arasında iyi bir geçişin sağlanmasıdır. Bu özellik, algoritmanın esnekliğini önemli ölçüde arttırmaktadır.
6. Genetik Algoritma ile optimizasyon için, her probleme özel bir optimizasyon tekniği kullanılması söz konusu değildir. Optimizasyonu gerçekleştirmek için boyutlandırma değişkenlerini, değişkenlerin değer uzayını ve problemin uygunluk fonksiyonunu belirlemek yeterlidir.
7. Alışılmış optimizasyon tekniklerine göre yapılan çözümlerde boyutlandırma değişkenlerinin ilk değerlerine bağlı olarak, algoritma her zaman sonuç vermeyebilir. Genetik Algoritma ile optimizasyonda ise, böyle bir sorunla karşılaşılsa dahi, bir sonraki adımda çözüm için uygun olmayan değerler grubu çözümden çıkartılmaktadır.

Bir Genetik Algoritma uygulaması, rastgele oluşturulmuş bireylerden teşkil edilen bir toplum ile başlar. Başlangıçtaki toplumun her bir bireyine ait kromozom, temsil ettiği veri yapısına uygun şekilde çözümlenerek problemin bağımsız değişkenleri elde edilir. Elde edilen bu değişkenler uygunluk fonksiyonunun parametreleri olarak alınır ve uygunluk değerleri hesaplanır. Bunun genetik bilimindeki karşılığı, canlının ortamın şartlarına uyum sağlamasının ölçüsüdür. Genetik Algoritmalarındaki anlamı ise, kromozom ile temsil

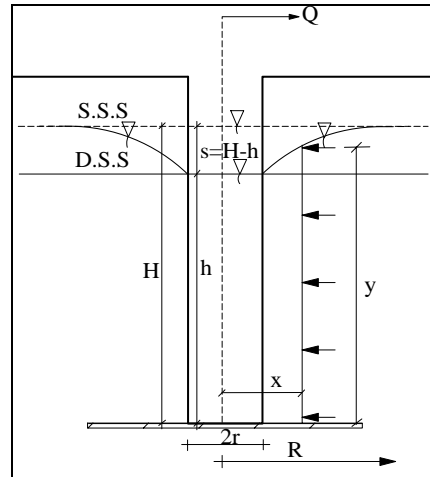
edilen değişkenlerin arzu edilen çözüme yakınlığıdır.

5. 4. Sayısal Örnekler

Bu çalışmada, su kaynakları sistemlerindeki lineer olmayan programlama problemleri Genetik Algoritma yöntemi kullanılarak çözülmüştür. Sınırlayıcısız lineer olmayan programlama problemi ve sınırlayıcılı lineer olmayan programlama problemi olmak üzere iki değişik örnek seçilmiştir. Çalışma daha sonra yapılacak bir dizi araştırma için bir ilk düzey çalışmasıdır. Bu nedenle kolay anlaşılabilmesi için basit örnekler seçilmiştir. Algoritma IBM uyumlu bilgisayarlara yönelik olarak BASIC programlama dilinde kodlanmıştır.

5. 4. 1. Sınırlayıcısız Lineer Olmayan Problem

Şekil 3'de görülen basınçsız bir napta açılan kuyunun akifer kalınlığı $H = 30$ m, permeabilite katsayısı $k = 0.0009$ m/sn ve yarıçapı $r = 0.25$ m dir. Bu çalışmada bu kuyudan çekilebilecek optimum debi bulunacaktır (Baylar ve ark., 1997).



Şekil 3. Serbest yüzeyli su yatağında açılmış hazneli kuyu

Dupuit hipotezlerine göre serbest yüzeyli yeraltı su tabakası içinde açılmış bir kuyuya zeminin vereceği debi Q_z ve hazneli kuyudan çekilmesine müsaade edilen en büyük debi Q_k debi değerlerinin birbirine eşit olduğu andaki debi optimum debi miktarı olarak adlandırılmaktadır. Optimum debiden daha fazla su çekilmesi durumunda, kuyuyu besleyen akifer tabakasının zemin daneleri akmaya başlamakta ve kuyu bir zaman sonra servisten çıkmaktadır. Optimum işletme debisinden daha az miktarda su alındığı zaman işletme masrafları artmaktadır.

Bu nedenle kuyuyu mümkün mertebe optimum işletme debisiyle veya buna yakın değerlerde işletmek gerekmektedir.

Hazneli kuyudan alınacak optimum debi miktarının saptanması için amaç fonksiyonu,

$$\text{Min } Z = \left| Q_z - Q_k \right| \pi k \frac{H^2 - h^2}{\ln \left(\frac{3000(H-h)\sqrt{k}}{r} \right)} - 2\pi r h \frac{\sqrt{k}}{15} = (5.1)$$

şeklinde yazılabilir.

Hazneli kuyulardan elde edilecek optimum debinin G. A. ile hesaplanması işlemi en önemli adım kromozom haritasını ve değerlendirme fonksiyonunu çok iyi bir şekilde oluşturmaktır. Çünkü eksik kromozom uzunluğu veya akılcı bir şekilde seçilmemiş değerlendirme fonksiyonu doğru sonuca ulaşamamaktadır.

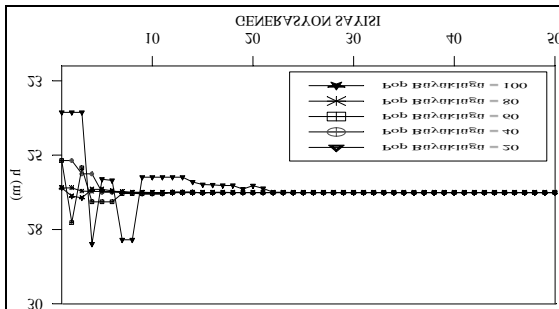
Sunulan çalışmada, h su yüksekliği değişken olarak seçilmiştir. İkilik tabandaki kromozom, on tabanına çevrilerek h değeri elde edilmektedir. Değerlendirme formülasyonu olarak bir penaltı fonksiyonu tanımlanmıştır.

$$PA = |Q_k - Q_z| \quad (5.2)$$

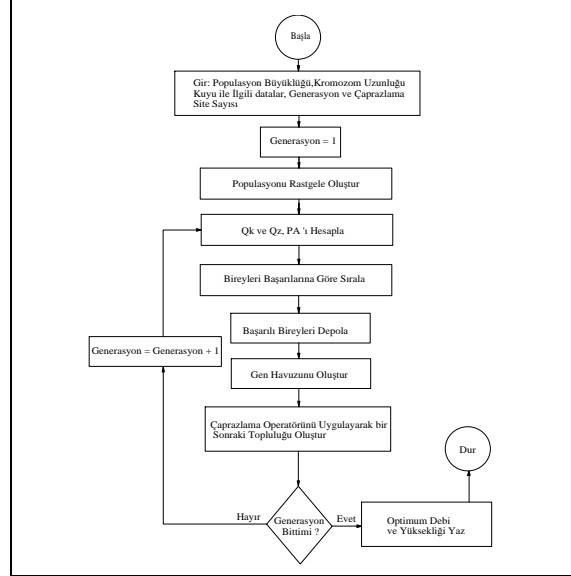
Burada amaç PA değerini minimize ederek, Q_k ve Q_z değerlerini birbirine eşitleyecek h değerini tespit etmektir.

$$\text{Amaç Fonksiyonu : Minimize } |Q_k - Q_z| \quad (5.3)$$

Yapılan bu çalışmada populasyon büyüklüğü 16, kromozom uzunluğu 15, generasyon sayısı 50 ve çaprazlama site sayısı da 3 seçilmiştir. Şekil 4'te görüldüğü gibi populasyon büyüklüğü, 20, 40, 60, 80, 100 olarak sırasıyla değiştirilerek populasyon büyüklüğünün G.A.'nın performansına etkisi araştırılmıştır. Populasyon sayısının büyük seçilmesi G.A.'nın performansını önemli derecede arttırmaktadır.



Şekil 4. Populasyon değişiminin genetik algoritmanın performansına etkisi
Genetik Algoritma işlemlerine ait akış diyagramı Şekil 5'te verilmektedir.



Şekil 5. Hazneli kuyulardaki optimum debinin bulunması için akış diyagramı

Yukarıda belirtilen veriler programa girildikten sonra G.A. artık 50 generasyondan oluşan evrim sürecini başlatacaktır. Evrim başlangıcında topluluk rastgele oluşturulmaktadır. Buna ait değerler Tablo 1'de verilmektedir.

Tablo 1, 1. ve 50. generasyona ait hesaplamaları göstermektedir. B. N. ile birey numarası verilmektedir. 2. kolonda eski populasyon olarak isimlendirilen rastgele oluşturulmuş topluluk gösterilmektedir. Bu topluluğa ait h değeri ve buna bağlı olarak da Q_z ve Q_k değerleri hesaplanmaktadır. Yine aynı kısımda bireyler başarı değerine göre yani, PA küçükten büyüğe doğru sıralanmaktadır. 6. kolonda ise kopyalama işlemi yapılmaktadır. Bu kısımda başarısız olan bireylerin yerini başarılı olan bireyler almaktadır. Örneğin burada kopyalama sayısı 1 seçilmiştir. Bu değer 1'den de büyük seçilebilir. 2. kolonda görüldüğü gibi en başarılı birey 1 nolu birey, en başarısız ise 16. bireydir. 6. kolonda görüldüğü gibi 16. birey gen havuzunun dışına atılmış ve onun yerine 1. birey kopya edilmiştir. Çünkü G. A.'nın amacı topluluktaki başarısız bireyleri eleyip başarılı olanları çoğaltmaktır. Böylelikle toplumun başarısı artırılabilir. Kopyalama işlemi bitirildikten sonra 7. kolondaki eşleştirme işlemi başlayacaktır. Çünkü bu işlem bireylerin eşleştirilmesi için gerekmektedir. Örneğin 1 nolu birey 9'nolu bireyle, 2'nolu birey

10' nolu bireyle eşleştirilmiştir. Bu eşleştirme işlemi rastgele yapılmaktadır. 8. kolonda ise bireylerin eşleştirme işlemi yapılmaktadır. Bu çalışma da çaprazlama site sayısı 3 olarak seçilmiştir. Çaprazlama noktaları da rastgele seçilmektedir. Çaprazlamanın G. A.'nın başarısına olan katkısı oldukça fazladır. Bu operatör sayesinde bireyler arasında genetik bilgi alış verişi sağlanmaktadır. Örneğin 6. kolondaki 2 ve 10 nolu elemanlar eşleştirildikten ve 15, 2 ve 7 nolu sitelerde çaprazlama işlemine tabi tutulduktan sonra 9. kolondaki bireyler oluşturulmaktadır. İkinci generasyonda yeni oluşturulan topluluk tekrar değerlendirilmekte ve başarı sıralaması

yapılmaktadır. Yeni topluluğun ismi ise eski topluluk olarak değiştirilmektedir. Bütün bu işlemler daha önceden verilmiş bir generasyon sayısına kadar devam etmektedir. h değeri H değerinden büyük olduğu zaman P_A büyük bir sayı alınarak bu birey gen havuzunun dışına atılmış ve Tablo 1'de görüldüğü gibi bu birey yerine (*) işareti konulmuştur.

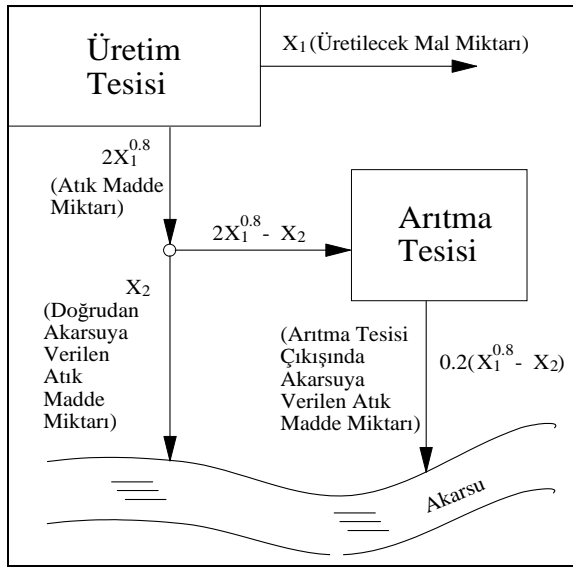
Tablo 1'de görüldüğü gibi topluluğun başarısı her bir generasyon da biraz daha artmakta ve bireyler en son 38 generasyonda birbirine benzemektedir. Bu aşamadan sonra bireyler artık çaprazlama operatörü ile değiştirilememektedir

Tablo 1. 1. - 50. Generasyona Ait Hesaplamalar

1. GENERASYON										
1	2	3	4	5	6	7	8			9
B.N.	Eski Populasyon	Q_2 (l/sn)	Q_k (l/sn)	h (m)	Kopyalama	Eşleştirme	Çaprazlama			Yeni Populasyon
1.	110101101001100	60.30	86.25	27.47	110101101001100	9	4	5	15	110111101001100
2.	110000100011101	105.92	78.06	24.86	110000100011101	10	15	2	7	10000000011100
3.	11011001011100	44.69	88.71	28.25	11011001011100	14	14	2	7	100111101011100
4.	111001011100110	18.49	92.36	29.41	111001011100110	16	7	12	3	110001111101110
5.	111001111011011	11.92	93.13	29.66	111001111011011	13	11	1	13	011001111001011
6.	11101000001010	10.53	93.28	29.71	11101000001010	11	5	11	6	11101000011010
7.	10000010000000	208.89	51.85	16.51	10000010000000	15	1	13	2	110000010000100
8.	011110101001000	216.18	49.26	15.69	011110101001000	12	13	4	12	011010101000000
9.	010110000101010	247.55	35.50	11.31	010110000101010	1	4	5	15	010100000101010
10.	001101000010000	267.56	20.95	6.67	001101000010000	2	15	2	7	011101100010001
11.	001010010010100	271.04	16.54	5.27	001010010010100	6	5	11	6	001010010000100
12.	001000001100001	272.94	13.17	4.19	001000001100001	8	13	4	12	00100001101001
13.	000011110101010	274.69	6.16	1.96	000011110101010	5	11	1	13	100011110111010
14.	000001111001000	274.54	3.04	0.97	000001111001000	3	14	2	7	010001011001000
15.	111110110101101	*	*	*	111110110101101	7	1	13	2	101110110101001
16.	111100010011101	*	*	*	110101101001100	4	7	12	3	111101001000100
2. GENERASYON										
1.	110001111101110	94.17	80.33	25.58	110001111101110	13	14	5	13	110001111101000
2.	11000010000100	108.30	77.58	24.71	11000010000100	14	11	13	10	110000010100000
3.	101110110101001	119.56	75.29	23.98	101110110101001	9	3	5	6	101101110101001
4.	110111101001100	39.63	89.46	28.49	110111101001100	15	12	11	7	110111001000100
5.	111010000011010	10.24	93.33	29.72	111010000011010	12	6	1	11	01101000001010
6.	100111101011100	168.77	63.79	20.32	100111101011100	11	1	7	9	000111101011100
7.	10001110111010	190.73	57.66	18.36	10001110111010	16	2	11	13	110011110101110
8.	100000000011100	209.81	51.53	16.41	100000000011100	10	15	7	14	100000100011100
9.	011101100010001	220.98	47.48	15.12	011101100010001	3	3	5	6	011110100010001
10.	011010101000000	232.49	42.80	13.63	011010101000000	8	15	7	14	011010001000000
11.	011001111001011	235.13	41.63	13.26	011001111001011	6	1	7	9	111001111001011
12.	010100000101010	253.14	32.29	10.28	010100000101010	5	6	1	11	110100000111010
13.	010001011001000	259.59	27.96	8.90	010001011001000	1	14	5	13	010001011001110
14.	001100001101001	268.73	19.62	6.25	001100001101001	2	11	13	10	001100001001101
15.	001010010000100	271.09	16.49	5.25	001010010000100	4	12	11	7	001010110001100
16.	111101001000100	*	*	*	110001111101110	7	2	11	13	100001111111010
50. GENERASYON										
1.	110011010011000	82.47	82.47	26.26	110011010011000	9	2	14	12	110011010011000
2.	110011010011000	82.47	82.47	26.26	110011010011000	14	14	1	5	110011010011000
3.	110011010011000	82.47	82.47	26.26	110011010011000	12	6	11	3	110011010011000
4.	110011010011000	82.47	82.47	26.26	110011010011000	16	15	10	11	110011010011000
5.	110011010011000	82.47	82.47	26.26	110011010011000	13	3	4	13	110011010011000
6.	110011010011000	82.47	82.47	26.26	110011010011000	11	3	6	15	110011010011000
7.	110011010011000	82.47	82.47	26.26	110011010011000	15	11	5	3	110011010011000
8.	110011010011000	82.47	82.47	26.26	110011010011000	10	2	6	5	110011010011000
9.	110011010011000	82.47	82.47	26.26	110011010011000	1	2	14	12	110011010011000
10.	110011010011000	82.47	82.47	26.26	110011010011000	8	2	6	5	110011010011000
11.	110011010011000	82.47	82.47	26.26	110011010011000	6	3	6	15	110011010011000
12.	110011010011000	82.47	82.47	26.26	110011010011000	3	6	11	3	110011010011000
13.	110011010011000	82.47	82.47	26.26	110011010011000	5	3	4	13	110011010011000
14.	110011010011000	82.47	82.47	26.26	110011010011000	2	14	1	5	110011010011000
15.	110011010011000	82.47	82.47	26.26	110011010011000	7	11	5	3	110011010011000
16.	110011010011000	82.47	82.47	26.26	110011010011000	4	15	10	11	110011010011000

5. 4. 2. Sınırlayıcı Linear Olmayan Problem

Şekil 6'da üretim ve arıtma tesisinden oluşan bir sistem görülmektedir. Üretim tesisinde üretilen malın birim miktarı (X_1) için $2X_1^{0.8}$ miktarda atık madde oluşmaktadır. Bu oluşan atık maddenin bir kısmı arıtma tesisinde işlenmeksizin doğrudan doğruya akarsuya (X_2), diğer kısmı da arıtma tesisinde işlenerek akarsuya verilmektedir. Akarsuyun su kalitesinin belli sınırlar içerisinde kalması istenmektedir. Bu çalışmada, üretim tesisinin toplam net kârını maksimum yapmak ve istenilen su



Şekil 6. Üretim-Arıtma tesisi sisteminin şematik gösterimi

kalitesini sağlamak için üreteceği mal miktarı ve akarsuya doğrudan doğruya arıtma tesisine göndermeksizin ne kadar atık madde verebileceği bulunacaktır. Sistem için amaç fonksiyonu ve sınırlayıcılar aşağıda verilmektedir (Mays and Tung, 1992).

5. 4. 3. Amaç Fonksiyonu

$$\text{Max } X_0 = -2X_1^{0.8} + 7X_1 - X_2$$

5. 4. 4. Sınırlayıcılar

$$2X_1^{0.8} - X_2 \leq 10$$

$$0.4X_1^{0.8} - 0.8X_2 \leq 4$$

$$2X_1^{0.8} - X_2 \geq 0$$

$$X_1 \geq 0 \text{ ve } X_2 \geq 0$$

Bu çalışmada; değişken olarak, X_1 ve X_2 seçilmiştir. İkili tabandaki kromozom, on tabanına çevrilerek X_1 ve X_2 nin değerleri elde edilmiştir. Populasyon büyüklüğü 14, kromozom uzunluğu 22 (ilk 11'i X_1 son 11'i ise X_2 'yi temsil etmektedir), generasyon sayısı 80 ve çaprazlama site sayısında 3 seçilmiştir. Problemden X_1 ve X_2 değerleri sınırlayıcıları sağlamayan zaman $X_0 = 0$ alınmış ve bu birey gen havuzunun dışına atılmıştır. Tablo 2'de 6 ve 13 nolu kromozomlara ait çaprazlama sitelerinin seçilmesi görülmektedir. Tablo 3'de ise 1 - 80. generasyona ait hesaplamalar görülmektedir.

Tablo 2. 6 ve 13'nolu Kromozomlara Ait Çaprazlama Sitelerinin Seçilmesi

GENLER																						
B.N.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
6	0	1	0	1	0	1	0	0	1	1	0	0	0	0	0	0	0	1	1	0	0	0
13	0	0	0	1	0	1	0	1	1	1	0	0	0	1	0	0	0	1	1	0	1	0

GENLER																						
B.N.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
6	0	1	0	1	0	1	0	1	1	1	0	0	0	0	0	0	0	1	1	0	0	0
13	0	0	0	1	0	1	0	0	1	1	0	0	0	1	0	0	0	1	1	0	1	0

6. TARTIŞMA VE SONUÇ

Genetik Algoritma ile optimizasyon, canlılardaki biyolojik ve sosyal işleyişin örneklenmesi ile elde edilmektedir. Hayali bir biyolojik topluluk ve bu topluluğun geçirdiği evrim sürecini incelemek Genetik Algoritmanın işleyişini somutlaştırmak açısından iyi bir yaklaşımdır.

Sunulan çalışma, bu konu üzerindeki bir seri çalışmanın ilk safhasıdır. Bu çalışmada, lineer olmayan programlama problemlerinin çözümünde

Genetik Algoritma tekniği kullanılmıştır. Bu maksatla, Genetik Algoritma esaslı bir optimizasyon programı geliştirilmiştir. Bu program ile lineer olmayan programlama problemlerinin Genetik Algoritma ile hassas olarak çözülebileceği görülmüştür.

Ayrıca çok amaçlı optimizasyon problemleri, dinamik programlama problemleri ve klasik matematiksel yöntemlerle çözümü mümkün olmayan çeşitli problemlerin Genetik Algoritmalar yardımıyla çözülebilmesi için çalışmalar devam etmektedir.

Tablo 3. 1. - 80. Generasyona Ait Hesaplamalar

1. GENERASYON										
1	2	3	4	5	6	7	8	9		
B.N.	Eski Populasyon	X ₁	X ₂	X ₀	Kopyalama	Eleştirme	Çaprazlama	Yeni Populasyon		
1.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	11	10	1	21	010101011000010011000
2.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	14	11	4	20	010101011000010011000
3.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	8	5	15	21	010110111000010011000
4.	010001011000010011000	9.30	2.00	51.19	010001011000010011000	10	1	18	10	110001011000010011000
5.	010100011000010011000	9.22	2.00	50.71	010100011000010011000	9	11	12	8	010100011000010011000
6.	010101001100000011000	8.10	1.92	44.12	010101001100000011000	13	8	18	1	010101011000000011000
7.	1101010011000010011000	8.11	2.00	44.10	1101010011000010011000	12	13	10	4	1101010011000010011000
8.	110100011000010011000	7.95	2.00	43.15	110100011000010011000	3	5	15	21	110100011000010011000
9.	1101000011000010011000	7.79	2.00	42.20	1101000011000010011000	5	11	12	8	110100011000010011000
10.	1101010001000110011000	5.55	2.04	28.93	1101010001000110011000	4	1	18	10	0101010001000110011000
11.	0101010001000010011000	5.54	2.00	28.91	0101010001000010011000	1	10	1	21	0101010001000010011000
12.	110100111000010001001	9.71	11.60	0.00	110100111000010001001	7	13	10	4	110100111000010001001
13.	000101011000100011010	9.36	7.08	0.00	000101011000100011010	6	8	18	1	000101011000100011010
14.	110101011000010001000	9.39	1.36	0.00	010101011000010001000	2	11	4	20	010101011000010001000
3. GENERASYON										
1.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	13	5	20	1	010101011000010011000
2.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	8	7	5	22	010110111000010011000
3.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	14	12	2	9	010101011000010011000
4.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	9	18	4	22	010101011000010011000
5.	010101011000010011000	9.38	2.00	51.67	010101011000010011000	11	14	21	6	01010001100010011000
6.	110001011000010011000	9.31	2.00	51.25	110001011000010011000	12	22	20	7	110001111000010011000
7.	1101010011000010011000	8.11	2.00	44.10	1101010011000010011000	10	20	15	8	1101010011000010011000
8.	010100011000010011000	7.94	2.00	43.09	010100011000010011000	2	7	5	22	010100011000010011000
9.	1101000011000010011000	7.79	2.00	42.20	1101000011000010011000	4	18	4	22	1101000011000010011000
10.	0101010001000010011000	5.54	2.00	28.91	0101010001000010011000	7	20	15	8	0101010001000010011000
11.	0101000001000110011000	5.22	2.04	27.00	0101000001000110011000	5	14	21	6	0101000001000110011000
12.	110100111000010001001	9.71	11.60	0.00	110100111000010001001	6	22	20	7	110100111000010001001
13.	0001010011000100011010	8.08	7.08	0.00	0001010011000100011010	1	5	20	1	1001010011000100011010
14.	010110111000010011000	9.54	2.00	0.00	110101011000010011000	3	12	2	9	110101011000010011000
80. GENERASYON										
1.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	9	4	20	3	110101011000010011000
2.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	13	18	20	16	110101011000010011000
3.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	11	2	18	7	110101011000010011000
4.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	8	4	17	21	110101011000010011000
5.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	10	12	7	17	110101011000010011000
6.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	14	9	7	21	110101011000010011000
7.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	12	11	2	14	110101011000010011000
8.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	4	4	17	21	110101011000010011000
9.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	1	4	20	3	110101011000010011000
10.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	5	12	7	17	110101011000010011000
11.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	3	2	18	7	110101011000010011000
12.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	7	11	2	14	110101011000010011000
13.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	2	18	20	16	110101011000010011000
14.	110101011000010011000	9.39	2.00	51.73	110101011000010011000	6	9	7	21	110101011000010011000

7. SEMBOLLER

- h : Hazneli kuyudaki su derinliği
H : Yeraltı suyu tabakasının kalınlığı
k : Permeabilite
PA : Penaltı fonksiyonu
r : Hazneli kuyu yarıçapı
R : Tesir yarıçapı
Q_k : Hazneli kuyudan çekilmesine müsaade edilen en büyük debi
Q_z : Dupuit hipotezlerine göre serbest yüzeyli yeraltı su tabakası içinde açılmış bir kuyuya zeminin vereceği debi

8. KAYNAKLAR

Arslan, A., Turgut, P. and Calayır, Y. 1996. "A Genetic Search Based Arrangement of Load Combinations in Structural Frames", CST 96, **The Third Int. Conf. On Computational Structures**

Technology, Budapest, Hungary, August 1996, Ed. B. H. V. Topping.

Baylar, A., Kaya, N. ve Arslan, A. 1997. "Kuyulardan Alınabilecek Optimum Debinin Hesaplanmasında Genetik Algoritma Yaklaşımı", **Su ve Çevre Sempozyumu**, TMMOB Jeoloji Mühendisleri Odası İstanbul Şubesi, 2-5 Haziran 1997, İstanbul.

Goldberg, D. E. 1983. Computer-Aided Gas Pipeline Operation Using Genetic Algorithms and Rule Learning, Ph.D Dissertation University of Michigan, Ann Arbor.

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

Mays, L. W., Tung, Y. K. 1992. Hydrosystems Engineering and Management, Mc Graw - Hill Series in Water Resources and Environmental Engineering..