

GENETİK, PARÇACIK SÜRÜ VE NELDER-MEAD SİMPEKS ALGORİTMALARININ FIRLATMA YÖRÜNGESİ OPTİMİZASYONU ÜZERİNDEN KARŞILAŞTIRILMASI

Hilal Arslan¹
Bakırçay Üni., İzmir

Ahmet Sofyalı² ve Emre Ünal³
Roketsan A.Ş., İstanbul ve Ankara

ÖZET

Bu bildiride, gerçekleştirilebilir ve optimal bir fırlatma yörüngesi elde etmek amacıyla, güdüm parametrelerinin optimizasyonu problemi, 3 serbestlik dereceli benzetim ile ele alınmıştır. Bu problemde, arama işlemi, arama uzayının büyüklüğü dolayısıyla son derece yüksek öneme sahiptir. Deneme-yanılma yaklaşımı, olası uygulamaların maliyetini hesaplayarak bütünsel (İng. global) optimal çözüm noktasının bulunmasında yetersizdir. Bütünsel optimal çözümün bulunması adına, genetik, parçacık sürü ve Nelder-Mead simpleks algoritmaları kullanılmış ve yöntemlerin başarımları karşılaştırmalı olarak sunulmuştur. Farklı hedef yörüngelere ulaşmak için kullanılan algoritmalarla çeşitli maliyet fonksiyonları da denenmiştir.

GİRİŞ

Yörünge optimizasyonu probleminde deterministik maliyet fonksiyonu (MF), alt ve üst sınırları bulunan problem değişkenlerinin tanım aralığında, belirli koşullar sağlanarak asgari kılınır. Optimizasyon problemlerini çözecek çok sayıda yöntem olmasına rağmen, güncel araştırmalar türevsiz ve biyolojik esinli yöntemlerin avantajlarını göstermiştir. Bu çalışmada, genetik algoritma (GA), parçacık sürü optimizasyonu (PSO) algoritması ve Nelder-Mead (NM) simpleks yöntemlerinin, fırlatma yörüngesi optimizasyonundaki başarımlar karşılaştırması yapılacaktır.

Holland [Holland, 1975] tarafından tanıtılan GA, doğal seçim ve genetikten türetilen evrimsel bir algoritmadır. “En güçlüünün hayatta kalması” yasası ile bireyleri seçip gelecek kuşakları üreterek, sürekli ve süresiz değişkenli, doğrusal olmayan MF’leri barındıran karmaşık tasarım problemlerine çözüm getirir. GA ilk olarak, probleme olası çözüm olabilecek rastgele bir kromozom nüfusu oluşturur. Sonrasında çözümün doğruluğuna karşılık gelen kromozomlar, MF ile değerlendirilir. Son olarak sonraki neslin oluşması için çaprazlama, mutasyon ve seçim işlemleri yapılır.

Kennedy ve Eberhart [Kennedy ve Eberhart, 1995] tarafından tanıtılan PSO yöntemi, GA’ya benzer şekilde nüfus temelli, biyolojik esinli bir yöntemdir. Yöntem, bir topluluğun topluluktaki en iyi üyeyi takip etme eğilimini de kapsayan sosyal davranış gözlemlerini temel alır. Topluluk üyelerinin başlangıç konumları, GA’da olduğu gibi rasgele bir işlemle belirlenir. Her bir parçacık, bireysel ve topluluk tecrübesi ile ilişkili bir hareket hızına sahip olarak arama uzayını tarar [Valle, Member, Venayagamoorthy, Member ve Harley, 2008]. Her bir parçacığın bir uyum değeri vardır ve en yüksek uyum değerine sahip olan parçacık, nüfusun bütünsel çözüm noktası olarak adlandırılır. Parçacıkların konum ve hızlarına göre güncellenen uyum değerleri ile arama uzayındaki bir sonraki konumlar belirlenir.

¹ Bilgisayar Mühendisliği Bölümü, Dr. Öğretim Üyesi, E-posta: hilal.arslan@bakircay.edu.tr

² Uçuş Mekanik Tekn. Birimi, İleri Teknolojiler ve Sistemler Grubu, Kıld. Uz. Müh., E-posta: ahmet.sofyalı@roketan.com.tr

³ Uçuş Mekanik Tekn. Birimi, İleri Teknolojiler ve Sistemler Grubu, Mühendis, E-posta: emre.unal@roketan.com.tr

NM simpleks yöntemi, Nelder ve Mead [Nelder ve Mead, 1965] tarafından tanıtılmış, bilinen en eski türevsiz optimizasyon yöntemlerinden biridir. Yerel bilgiden faydalanarak en yakın optimum noktaya yönelir. Yerel minimum noktasını arayan yöntem, çok boyutlu optimizasyon problemlerinde kullanılabilir. Gradyan yöntemleri gibi türev kullanarak, bir fonksiyon üzerinde hareket etmek zorunda değildir. Simpleksin köşelerinde MF hesaplayarak, maliyeti en yüksek olan, en kötü köşeyi yeni bir köşe ile değiştirir. Bu değiştirme işlemi dört temel uygulama ile gerçekleştirilir: yansıma, genişleme, kısılma ve büzülme. Bu işlem, simpleks yeteri kadar küçülüp yerel minimum noktası bulunana kadar tekrarlanır.

Bu bildiri şu şekilde düzenlenmiştir: Ele alınan problemle ilgili çalışmalar izleyen bölümde incelenmiştir. Fırlatma yörüngesi optimizasyonunun uygulama ayrıntıları sonraki bölümde sunulmuştur. Son iki bölümdeyse, optimizasyon sonuçları tartışılmış ve bildiri sonuçlandırılmıştır.

İLGİLİ ÇALIŞMALAR

Son yıllarda, yörünge optimizasyonu probleminin evrimsel algoritmalarla çözümünde kayda değer ilerleme gösterilmiştir. Hassan ve çalışma arkadaşları [Hassan, Cohanım, Weck, 2005], denek bir problem seti üzerinde PSO ve GA başarımlarını karşılaştırmışlar ve PSO'nun kısıtlanmamış doğrusal olmayan problemlerde GA'ya üstünlüğünü göstermişlerdir. Joseph ve Thomas [Joseph, Thomas, 2015], NM ve PSO yöntemlerini gen ifadeli veri kümelenmesinde değerlendirmişlerdir. PSO yönteminin kolayca yerel minimumlara takıldığını gösterdikleri çalışmalarında, NM ve PSO yöntemlerini bu zorluğu aşmak üzere birleştirmişlerdir. Hamid ve arkadaşları [Hamid, Rahim, Selveraj, 2016], NM ve PSO yöntemlerinin melez bir uygulamasını güneş hücresi ve fotovoltaiik modüllerinin parametre tanımlamasında kullanmışlardır. Fan ve arkadaşları [Fan, Liang, Zahara, 2006], NM yöntemini PSO ve GA ile birleştirerek, doğrusal olmayan ve sürekli değişen fonksiyonların bütünsel optimal çözümünü bulacak bir yöntem önermişlerdir. Çalışmalarında daha çok yanıt yüzey metodolojisine odaklanmışlardır. Testlerde karmaşık, doğrusal olmayan, sürekli fonksiyonları kullanarak, melez algoritmaların bütünsel optimal çözümü etkili şekilde bulunduğunu göstermişlerdir. West ve arkadaşları [West, Montazeri, Monk ve Taylor, 2016], 7 serbestlik dereceli robotik manipülatörün parametrelerini optimize etmek için bir GA yaklaşımı önermişlerdir. 2018 yılındaysa, Shirazi ve arkadaşları [Shirazi, Ceberio, Lozano, 2018], uzay aracı yörünge optimizasyonu problemlerinin çözümleri üzerine bir derleme yayımlamışlardır. Derlemede, optimizasyon problemleri çözümleri üzerinde dört adıma odaklanmışlardır: Problemin matematiksel olarak modellenmesi, MF tanımlaması, çözüm yönteminin geliştirilmesi ve çözüm. Çalışmada her bir adım açıklanarak, nasıl karar verileceği ve seçim yapılacağına ilişkin ayrıntılı bilgi verilmiştir.

FIRLATMA ARACI YÖRÜNGE OPTİMİZASYONU

Bu bölümde fırlatma yörüngesi optimizasyonu problemi çözümünde uygulanan yaklaşım

- İlerletme (İng. propagation), araç ve ortam modelleri,
- maliyet fonksiyonları,
- çözüm yöntemleri

verilerek sunulacaktır.

Kullanılan Modeller

Başlangıç konumu basık Dünya üzerinde, jeodezik enlem, boylam (Grinviç meridyeninden doğuya doğru pozitif ölçülür) ve (jeodezik) irtifa bilgileri ile tanımlanmıştır. İlk hız, bağıl hızın referansı olan Dünya-merkezli-Dünya'da-çakılı [İng. Earth-centered-Earth-fixed (ECEF)] eksen takımında, bağıl sürat ile her ikisi de basık Dünya yüzeyine göre hesaplanan uçuş hattı açısı (UHA) ve azimut açısı ile verilmiştir. İlk değer atamayı ve ilerletmeyi olumsuz etkileyecek olan tekilliklerden kaçınmak adına, sayısal entegrasyon, Dünya-merkezli-eylemsiz [İng. Earth-centered-inertial (ECI)] referans sisteminde yazılan bir durum vektörü üzerinden yapılmıştır. Ara referans sistemi olarak, küresel Dünya üzerinde tanımlı olan yerel-ufuksal-yerel-düşey [İng. local-horizontal-local-vertical (LHLV)] referans sistemi kullanılmıştır. LHLV referans sisteminde konum ve hız; sırasıyla; jeosantrik enlem, boylam ve yarıçap (Dünya'nın merkezinden ölçülür); bağıl sürat ile her ikisi de küresel Dünya yüzeyine göre hesaplanan UHA ve azimut açısıyla tanımlanır. Fırlatma aracı gövdesinin kütle

merkezi etrafındaki açısal hareketi, diğer deyişle yönelme hareketi, bu çalışmanın kapsamı dışında tutulmuştur.

Fırlatma aracı çok kademelidir ve itki profilleri, ateşleme sonrası ve yanma sonu geçiş evreleri hesaba katılarak gerçekçi şekilde modellenmiştir. Kütlesel ve aerodinamik karakteristikler orta seviyede ayrıntılandırılmıştır. Kademe ayrılmasının kütlesel ve aerodinamik etkileriye benzetimlerde hesaba katılmıştır. Benzetim zaman adımı 1 saniyedir.

Yer çekimi modelinde, basıklık etkisinin yalnızca en büyük bileşeni (J_2) değerlendirmeye alınmıştır. Bu durumda, yer çekimi ivme vektörü Dünya'nın geometrik merkezinden geçmez. Atmosferse, 0-1000 km'lik irtifa aralığında 1 km çözünürlükle modellenmiştir.

Fırlatma yörüngesinin elde edilmesinde aşağıdaki basit ve göreceli olarak noksan güdüm yaklaşımı kullanılmıştır:

- 1) Burun eğme (İng. pitch over): *Burun eğme* manevrası, *başlangıç Mach (BEBM)* değeriyle başlar ve *son Mach (BESM)* değeriyle biter; güdüm açısı, *burun eğme genişliğinde (BEG)* bir yarım sinüs dalgası biçiminde değişir.
- 2) İtkisiz uçuş: *İtkisiz uçuş* evresi, *başlangıç irtifasında (İUBİ)* başlar ve *son uçuş hattı açısına (İUSUHA)* erişildiğinde biter.
- 3) Yörüngeye oturtma: Yörüngeye oturtma manevrası, *itkisiz uçuş son uçuş hattı açısı (İUSUHA)* değerine erişildiğinde başlar ve UHA sıfır olduğunda veya üst kademenin yakıtı tükendiğinde biter; güdüm açısı, *yörüngeye oturtma genişliğinde (YOG)* bir sinüs dalgası biçiminde ve *yörüngeye oturtma faz açısıyla (YOFA)* değişir.

BEBM, BESM, BEG, İUBİ, İUSUHA, YOG ve *YOFA*'dan oluşan yedi güdüm parametresine ek olarak, *üst kademe toplam yanma süresi (ÜKTYs)* optimizasyonun sekizinci ve son parametresini oluşturur.

Üst kademenin uçuş evresine ait yarıçap, sürat ve UHA'nın benzetimin son anındaki değerleri, izleyen alt başlıkta incelenecek olan MF'lerde yerine konulur. Benzetim, UHA sıfırın altında bir değer aldığı anda veya üst kademenin yakıtı tükendiğinde sonlandırılmaktadır.

Kullanılan Maliyet Fonksiyonları

300 km'lik jeosantrik irtifalı bir dairesel yörüngeye oturma'nın hedeflendiği optimizasyon denemelerinde, asgari kılınmak üzere kullanılan ilk MF Denklem 1'de verilmiştir.

$$MF_1 = \text{maliyet}_1^2 + \text{maliyet}_2^2 + \text{maliyet}_3^2 \quad (1)$$

Burada; $\text{maliyet}_1 = (\text{sürat} - \text{sürat}_{300})/\text{sürat}_{300}$; $\text{maliyet}_2 = (\text{yarıçap} - \text{yarıçap}_{300})/\text{yarıçap}_{300}$; $\text{maliyet}_3 = \text{UHA}|_{\text{radyan}}$. MF_1 'i asgari kılmak, irtifa, sürat ve UHA'dan her birinin, hedef değerlerden mutlak sapmasını asgari kılmayı gerektirir.

İkinci MF aşağıdaki gibidir:

$$MF_2 = [(\text{maliyet}_1 \times \text{maliyet}_2) - 1]^2 + \text{maliyet}_3^2 \quad (2)$$

Buradaysa; $\text{maliyet}_1 = \text{sürat}/\text{sürat}_{300}$; $\text{maliyet}_2 = \text{yarıçap}/\text{yarıçap}_{300}$; $\text{maliyet}_3 = \text{UHA}|_{\text{radyan}}$. Bilindiği gibi, dairesel yörüngede yarıçap ile süratin çarpımı, sabit olan özgül açısal momentumun şiddetine (h_{300}) eşittir. MF_2 'de bulunan ilk terim, söz konusu çarpımın hedef değere ulaşmasını sağlamaya yöneliktir.

MF_3 ve MF_4 , yörünge geometrisini belirleyen iki yörünge parametresini temel alır: yarı büyük eksen uzunluğu (a) ve dışmerkezlik (e).

$$h = \text{sürat} \times \text{yarıçap} \times \cos(\text{UHA}) \quad (3a)$$

$$a = \frac{\mu_{\text{Dünya}} \times \text{yarıçap}}{(2 \times \mu_{\text{Dünya}}) - \text{sürat}^2 \times \text{yarıçap}} \quad (3b)$$

$$\varepsilon = -\frac{\mu_{Dünya}}{2 \times a} \quad (3c)$$

$$e = \sqrt{1 + \left(2 \times \frac{h^2}{\mu_{Dünya}} \times \frac{\varepsilon}{\mu_{Dünya}} \right)} \quad (3d)$$

Denklem 4'te verili MF_3 'ün nihayetinde asgari kılınması, hedef yörüngeye oturmayı garanti etmelidir.

$$MF_3 = maliyet_1^2 + maliyet_2^2 \quad (4)$$

Bu kez; $maliyet_1 = (a - a_{300})/a_{300} = (a - yarıçap_{300})/yarıçap_{300}$; $maliyet_2 = e - e_{300} = e$. Öte yandan, Denklem 5'te verili MF_4 kullanıldığında, en yüksek mekanik enerjiye sahip dairesel yörünge hedeflenir.

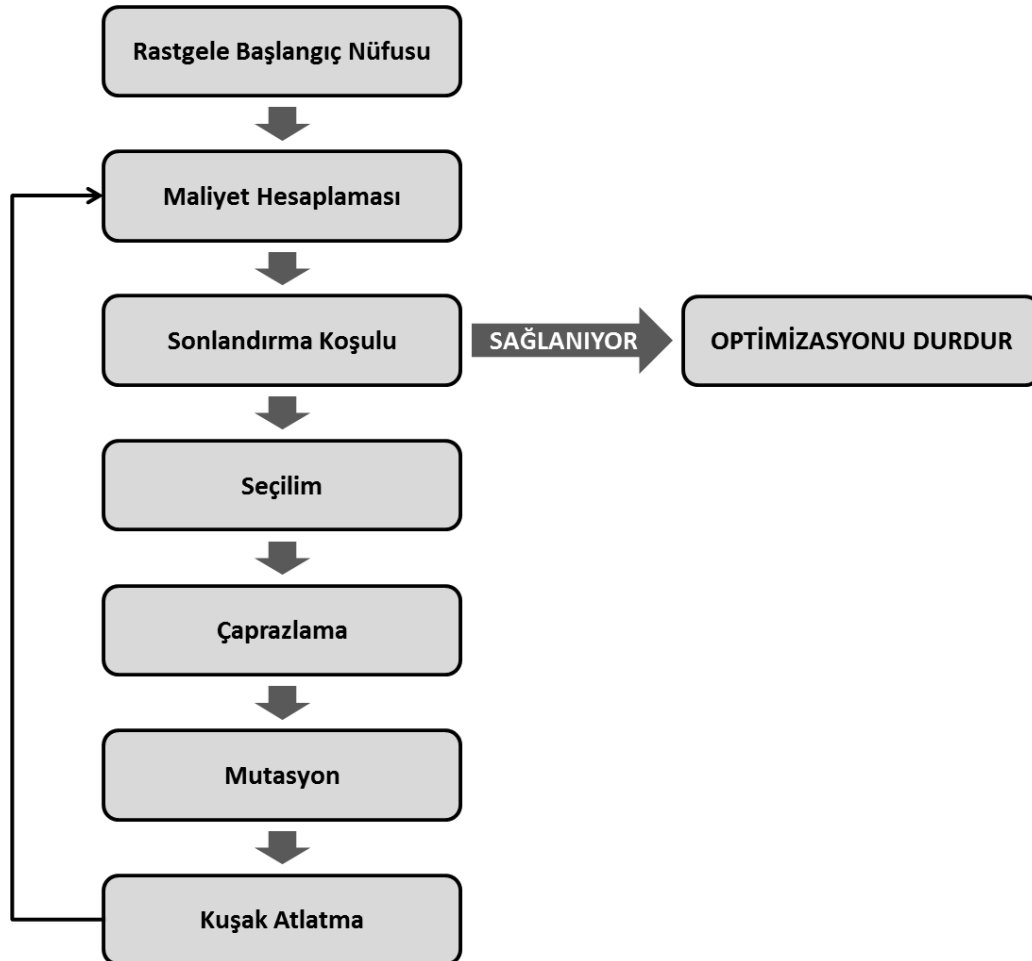
$$MF_4 = e/a \quad (5)$$

MF_4 'ün kullanıldığı optimizasyonun sonucunda, fırlatma aracında depolu kimyasal enerjinin, aracın mekanik enerjisine dönüştürülme oranı azami kılınmış olmalıdır.

Sıradaki alt başlıkta, GA, PSO ve NM simpleks yöntemlerinin optimal parametre değerlerinin bulunması amacıyla nasıl uygulandığı görülecektir.

Çözüm Yöntemi 1: Genetik Algoritma

Önceki alt başlıkta sunulan MF 'ler kullanılarak yörünge optimizasyonu için ilk olarak GA kullanılmıştır. Güdüm parametreleri her bir genin daha önce tanımlanmış aralıkta değişen bir gerçek sayıyı ifade ettiği sekiz gen olarak kodlanmıştır. GA'nın şematik gösterimi Şekil 1'de yer almaktadır.



Şekil 1. Genetik algoritmanın şematik gösterimi

Algoritma, aday çözümlerin bulunduğu rastgele bir nüfus oluşturularak başlatılır. Sonrasında, her bir kromozomun uyum değeri bulunarak ebeveyn bireyleri seçmek için rulet tekerleği yöntemi kullanılır. Çocuklar, seçilmiş bireylerin tek nokta çaprazlamalarıyla ve düzenli rastgele mutasyonlarıyla üretilir. Belirli bir sonlandırma şartı gerçekleşene kadar, çocuklar ebeveynlerin yerine geçirilerek birbirini izleyen kuşaklar oluşturulur. Yeni kuşak oluşturulurken, optimal çözüme yakınsamayı güçlendirmek adına elitizm kullanılmıştır. Sonlandırma kuşak sayısına bağlanmıştır.

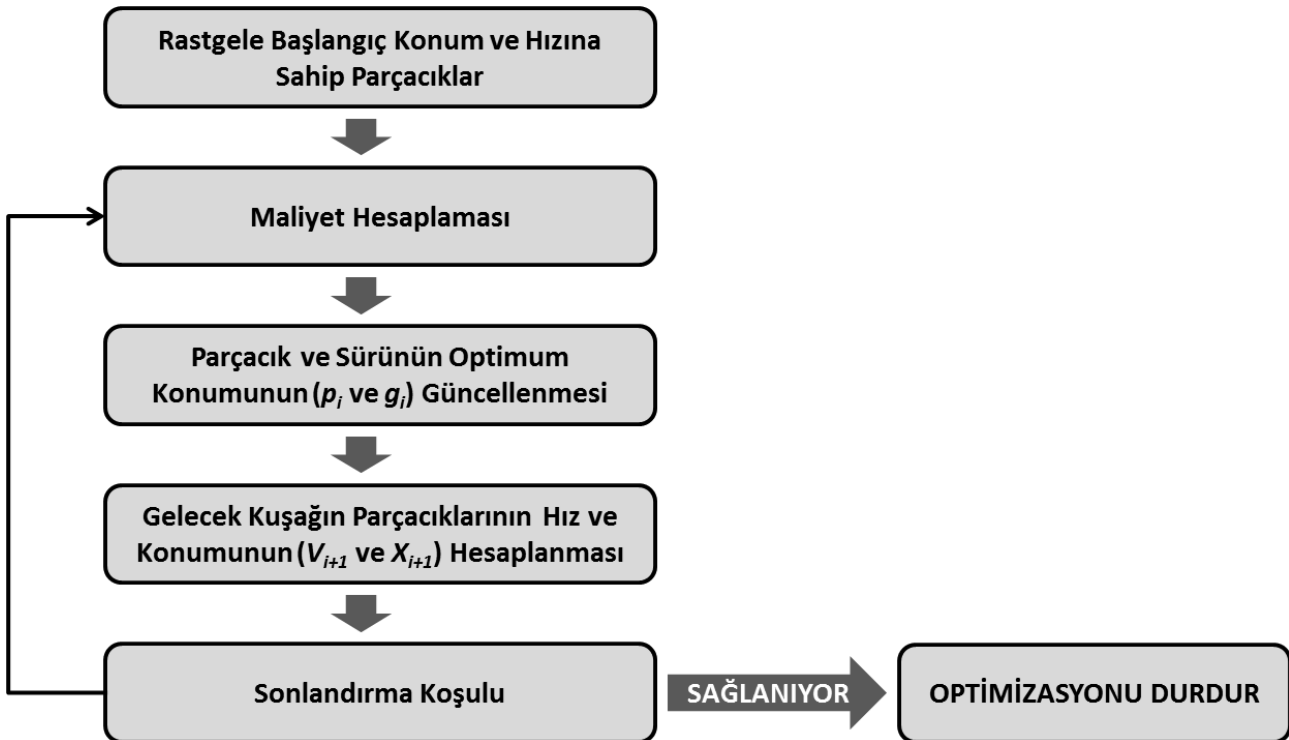
Çözüm Yöntemi 2: Parçacık Sürü Optimizasyonu

PSO algoritması, parçacık konumlarıyla temsil edilen parametrelerin optimal değerlerinin bulunması amacıyla ve GA'nın uygulandığıyla aynı problem için kullanılmıştır. İlk adımda, parçacıkların hız ve konumları rastgele olarak üretilir. Sonrasında, her bir parçacığın MF değeri hesaplanarak, konum ve hız değerleri Denklem 6 ve Denklem 7'ye göre güncellenir.

$$V_{i+1} = w * V_i + c_1 rand_1 * (p_i - X_i) + c_2 rand_2 * (g_i - X_i) \quad (6)$$

$$X_{i+1} = X_i + V_{i+1} \quad (7)$$

Burada; X konum vektörü, V hız vektörü, $rand_1$ ve $rand_2$ 0 ile 1 arasında rastgele üretilen birer gerçek sayı, p parçacığın bulunduğu optimum konum ve g sürünün bulunduğu optimum konumu simgelemektedir. Denklem 6'da görüldüğü gibi, her bir parçacığın mevcut hesaplama anındaki hızı, bir önceki andaki konum ve hızına, bulunduğu optimum konuma ve sürünün bulunduğu optimum konuma bağlıdır. Her bir parçacığın mevcut andaki konumuysa, bir önceki andaki konumu ile mevcut andaki hızına bağlıdır (bkz. Denklem 7). Sonlandırma şartı sağlanana dek, anlarda adım adım ilerlenerek optimal çözüme ulaşılır. Sonlandırma şartı, GA'daki kuşak sayısına karşılık gelen adım sayısı için tanımlanmıştır. PSO algoritmasının akış şeması Şekil 2'de gösterilmiştir.



Şekil 2. PSO algoritmasının şematik gösterimi

Çözüm Yöntemi 3: Nelder-Mead Simpleks Yöntemi

NM simpleks yöntemi, türevsiz ve hesaplamaları basit olan bir yöntemdir. Algoritma, dört tip operasyonu temel alır: Yansıma, genişleme, kısalma ve büzülme. Bunlar, sırasıyla, α , β , γ , δ değişkenleriyle, $\alpha > 0$, $\beta > 1$, $0 < \gamma < 1$, $0 < \delta < 1$ şartlarını sağlayacak şekilde temsil edilir. n boyutlu bir simpleks oluşturacak $n + 1$ köşe $(x_1, x_2, \dots, x_{n+1})$, kullanıcı tarafından atanır veya rastgele şekilde üretilir; böylelikle adım adım yineleme (İng. iteration) başlar [Gao ve Han, 2012].

Adım 1: $x_1, x_2, \dots, x_{n+1}$ köşeleri için maliyetler hesaplanır ve $f(x_1) < f(x_2) < \dots < f(x_{n+1})$ durumu sağlanacak şekilde sıralama yapılır. Burada $f(x_i)$ maliyet fonksiyonudur.

Adım 2: Ardından yansıma noktası, x_r ,

$$x_r = \bar{x} + \alpha(\bar{x} - x_{n-1}) \quad (8a)$$

şeklinde hesaplanır. Yansıma noktasına karşılık gelen maliyet, $f(x_r)$, hesaplanır. Eğer $f(x_1) < f(x_r) < f(x_{n+1})$ ise x_{n+1} , x_r ile yer değiştirir.

Adım 3: $f(x_r) < f(x_1)$ durumunda adım uzar ve bir genişleme noktası, x_e ,

$$x_e = \bar{x} + \beta(x_r - \bar{x}) \quad (8b)$$

şeklinde hesaplanır. Karşılık gelen maliyet, $f(x_e)$, bulunur ve eğer $f(x_e) < f(x_r)$ ise x_e ; aksi durumda x_r , x_{n+1} 'in yerini alır.

Adım 4: Eğer $f(x_n) < f(x_r) < f(x_{n+1})$ ise bir dış kısalma noktası, x_{oc} ,

$$x_{oc} = \bar{x} + \gamma(x_r + \bar{x}) \quad (8c)$$

şeklinde hesaplanır. Karşılık gelen maliyet, $f(x_{oc})$, belirlenir; $f(x_{oc}) \leq f(x_r)$ ise, x_{n+1} , x_{oc} ile yer değiştirir; değilse Adım 6'ya geçilir.

Adım 5: Son olarak $f(x_{n+1}) < f(x_r)$ durumunda bir iç kısalma noktası, x_{ic} ,

$$x_{ic} = \bar{x} - \gamma(x_r - \bar{x}) \quad (8d)$$

şeklinde hesaplanır. Karşılık gelen maliyet, $f(x_{ic})$, bulunur. $f(x_{ic}) < f(x_{n+1})$ durumunda, x_{ic} , x_{n+1} 'in yerini alır; aksi durumda Adım 6 uygulanır.

Adım 6: $2 \leq i \leq n + 1$ durumunda, tüm x_i değerleri

$$x_i = x_1 + \delta(x_i - x_1) \quad (8e)$$

işlemiyle büzülür. Belirlenen sonlandırma şartı sağlanana dek yineleme sürer.

Her bir yinelemede köşelerin değişmesiyle, simpleks yerel olarak asgari kılınır. Bütünsel olarak asgari kılınması mümkün, fakat düşük ihtimallidir. Yalnızca konveks yapıli problemlerde bütünsel optimuma (minimuma) yakınsama garanti edilir; aksi hallerde, yöntemin yerel bir optimuma (minimuma) takılması muhtemeldir.

OPTİMİZASYON SONUÇLARI

Bu bölümde, optimize edilmiş parametreler kullanılarak elde edilen yörüngeler, karşılık gelen MF değerleri ve çözüm süreleriyle birlikte sunulacaktır. Sekiz boyutlu arama uzayının alt ve üst sınırları Tablo 1'de görülmektedir.

Tablo 1: Optimizasyon parametreleri ve değişim aralıkları

Parametre	Değişim aralığı alt sınırı	Değişim aralığı üst sınırı
BEBM [-]	0,05	0,5
BESM [-]	0,05	0,5
BEG [derece]	1	16
İUBİ [m]	80000	110000 veya 120000
İUSUHA [derece]	1	15
YOG [derece]	-12	12
YOFA [derece]	-90	90
ÜKTYS [sn]	100	170

Tüm hesaplamalar, Intel Xeon CPU 8 x E5-1620 V4 3.5 GHz işlemciler ve 16 GB'lık belleğe sahip bir makineye yaptırılmıştır; makinenin işletim sistemi Windows 7'dir. GA yöntemi, C++ ortamında UNIX ve Windows işletim sistemlerini destekleyen, açık kaynak kodlu GA kütüphanesi olan GALib'den [Url-1] yararlanılarak uygulanmıştır. PSO ve NM yöntemleri ise, yine açık kaynak kodlu, doğrusal olmayan fonksiyonlar kütüphanesi olan OptimLib'den [Url-2] yararlanılarak uygulanmıştır; OptimLib, C++ dilinde yazılmış yerel ve bütünsel optimizasyon algoritmaları içermektedir.

Genetik Algoritma Uygulaması Sonuçları

GALib içindeki "Basic GA", turnuvayla seçim yöntemi ve tek nokta çaprazlamasıyla kullanılmıştır. Algoritmanın parametreleri Tablo 2'de özetlenmiştir.

Tablo 2 Genetik algoritma parametreleri

Kuşak sayısı	150
Nüfus	300
Çaprazlama olasılığı	1,0
Mutasyon olasılığı	0,5

GA ile elde edilen optimizasyon sonuçları Tablo 3'te gösterilmektedir.

Tablo 3 Genetik algoritmayla optimizasyon sonuçları

	MF ₂ ile	MF ₄ ile
Optimizasyon girdileri		
BEBM [-]	0,217	0,250
BESM [-]	0,413	0,416
BEG [der]	6,765	9,132
İUBİ [m]	93073,4	96347,9
İUSUHA [der]	10,09	11,37
YOG [der]	-0,687	-0,837
YOFA [der]	28,67	-3,782
ÜKTYS [sn]	119,0	125,1
Optimizasyon çıktıları		
MF	$3,729 \times 10^{-8}$	$3,147 \times 10^{-11}$
yarıçap [m]	6853130	6850360
sürat [m/sn]	7529,53	7627,85
UHA [der]	0,008	0,12
a [m]	6684265	6850051
e [-]	0,025	$2,158 \times 10^{-4}$
asgari jeosantrik irtifa [m]	137260,3	470435,5
azami jeosantrik irtifa [m]	474995,4	473391,9
Optimizasyon süresi [sn]	42278,2	38349,6

Parçacık Sürü Optimizasyonu Sonuçları

OptimLib içinde bulunan "Basic PSO" kullanılmıştır. Elde edilen sonuçlar Tablo 4'te sunulmuştur.

Tablo 4 Parçacık sürü optimizasyonu sonuçları

	MF ₁ ile	MF ₂ ile	MF ₃ ile	MF ₄ ile
PSO parametreleri				
Adım sayısı	50	50	50	50
Parçacık sayısı	300	300	300	300
Optimizasyon girdileri				
BEBM [-]	0,244	0,244	0,229	0,120
BESM [-]	0,486	0,459	0,489	0,390
BEG [der]	6,638	6,925	5,936	5,377
İUBİ [m]	109595	109992	119490	119812
İUSUHA [der]	12,46	12,67	12,29	13,12

YOG [der]	1,426	-0,121	1,498	-4,403
YOFA [der]	-79,60	-78,44	-83,57	-39,12
ÜKTYS [sn]	160,1	153,3	159,2	140,3
Optimizasyon çıktıları				
MF	$1,326 \times 10^{-4}$	$7,974 \times 10^{-8}$	$5,185 \times 10^{-5}$	$9,443 \times 10^{-11}$
yarıçap [m]	6745510	6773820	6745930	6825480
sürat [m/sn]	7683,65	7618,23	7667,32	7639,94
UHA [der]	0,060	0,011	0,044	0,022
a [m]	6739502	6682205	6711888	6821959
e [-]	0,001	0,014	0,005	$6,439 \times 10^{-4}$
asgari jeosantrik irtifa [m]	352098,9	212444,9	299329,7	439429,8
azami jeosantrik irtifa [m]	370630,3	395691,7	368172,8	448214,5
Optimizasyon süresi [sn]	36934,7	39712,1	39878,3	35878,8

Nelder-Mead Simpleks Uygulaması Sonuçları

OptimLib içinde bulunan NM simpleks yöntemi kullanılmıştır. Yöntemin kullandığı yansıma, genişleme, kısalma ve büzülme parametreleri, sırasıyla, $\alpha = 1$, $\beta = 2$, $\gamma = 0,5$ ve $\delta = 0,5$ olarak seçilmiştir. Elde edilen sonuçlar Tablo 5'te gösterilmiştir.

Tablo 5 Nelder-Mead simpleks uygulaması sonuçları

	MF ₁ ile	MF ₂ ile
Optimizasyon girdileri		
BEBM [-]	0,282	0,269
BESM [-]	0,297	0,283
BEG [der]	15,89	11,49
İUBİ [m]	110000	120000
İUSUHA [der]	14,87	10,64
YOG [der]	$2,839 \times 10^{-6}$	$4,967 \times 10^{-5}$
YOFA [der]	$3,101 \times 10^{-5}$	$4,960 \times 10^{-5}$
ÜKTYS [sn]	170	170
Optimizasyon çıktıları		
MF	0,092	$1,235 \times 10^{-7}$
yarıçap [m]	6902150	6899900
sürat [m/sn]	5208,42	5101,01
UHA [der]	1,085	0,856
a [m]	4510448	445275,6
e [-]	0,531	0,550
asgari jeosantrik irtifa [m]	-4260488	-4373158
azami jeosantrik irtifa [m]	525108,7	522394,5

SONUÇ

Bu çalışmada fırlatma yörüngesi optimizasyonu ele alınmıştır. Bu bildiri, genetik algoritma, parçacık sürü ve Nelder-Mead simpleks yöntemlerinin optimizasyon başarımlarını farklı maliyet fonksiyonları için ve karşılaştırmalı şekilde sunmaktadır.

Tablo 3 ve 4 göstermektedir ki, MF₁, MF₂ ve MF₃, hedeflenen irtifadaki dairesel bir yörüngeye yerleşmeye hizmet etmemektedir. Bu üç MF, çeşitli ve birden fazla, normalleştirilmiş, dolayısıyla boyutsuz terimin karesinin toplamına eşittir. Toplam halinde yazılı MF'lerden verim alınabilmesinin şartı, terimlerinin uygun şekilde ağırlıklandırılmış olmasıdır. Gelecek çalışmada, toplamı bire eşit olacak ağırlıklandırma çarpanları kullanılarak, üç MF ile de problemi çözecek çarpan setlerinin bulunması amacıyla deneme-yanılma yaklaşımından yararlanılacaktır, ki bu ek hesaplama yükü anlamına gelmektedir. Bu nedenle, MF tanımlamasının, optimizasyon yöntemi seçiminden, hatta optimizasyon parametrelerinin belirlenmesinden daha önemli olduğu söylenebilir. Ağırlıklandırma

gerekliliğinden kurtulmak adına, hangi problem ele alınırsa alınsın, tek terimli MF kullanımı hedeflenmelidir.

Tek terimli MF_4 kullanıldığında beklenen çözümün elde edilebildiği Tablo 3 ve 4'te görülmektedir. Bununla birlikte, GA'yla elde edilen sonuç PSO'nun verdiğiinden daha iyidir, bunda GA'nın kuşak sayısının PSO'nun adım sayısının üç katı olarak alınmış olmasının belirleyici bir payı vardır. Sonuç olarak, MF_4 , verili fırlatma aracıyla, yaklaşık 472 km'lik ortalama jeosantrik irtifalı ve neredeyse dairesel bir yörüngeye oturma'nın mümkün olduğunu ortaya koymayı sağlamıştır.

Tablo 5'te yer alan sonuçlar göstermektedir ki, NM yöntemi ele alınan optimizasyon problemini çözememektedir. Yöntemin yerel arama işlemlerinde çok etkili olduğu bilinse de, bir optimuma yakınsaması, seçilen başlangıç durumuna oldukça duyarlıdır.

Sonraki çalışmalarda, optimal fırlatma yörüngesi, PSO-NM yöntemi gibi hibrit yaklaşımlar probleme uygulanarak aranacaktır. Bu yapılırken, algoritmalar paralel olarak programlanarak optimizasyon süreleri asgari kılınmaya çalışılacaktır.

Kaynaklar

- Fan, S. S., Liang, Y., Zahara, E., 2006. *A genetic algorithm and a particle swarm optimizer hybridized with Nelder-Mead simplex search*, Computers & Industrial Engineering, Vol. 50, pp. 401-425
- Gao, F. and Han, L., 2012. *Implementing the Nelder-Mead simplex algorithm with adaptive parameters*, Computational Optimization and Applications
- Hamid, N. F. A., Rahim, N. A., and Selvaraj, J., 2016. *Solar cell parameters identification using hybrid Nelder-Mead and modified particle swarm optimization*, Journal of Renewable and Sustainable Energy, Vol. 8, pp. 015502
- Hassan, R., Cohanım, B., and Weck, O., 2005. *A Comparison of Particle Swarm Optimization and The Genetic Algorithm*, 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference, Austin, Texas
- Holland, J. H., 1975. *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor,
- Joseph, M. and Thomas, J., 2015. *Comparison of Nelder-Mead, Particle Swarm Optimization, and NM-PSO for Clustering of Gene Expression Data*, International Journal of Advanced Computing and Electronics Technology, Vol. 2, No. 2, pp. 2394-3416
- Kennedy, J., Eberhart, R., 1995. *Particle Swarm Optimization*, Proceedings of the ICNN'95-International Conference on Neural Networks, pp. 1942-1948
- Nelder, J.A., Mead, R. 1965. *A simplex method for function minimization*, Comput. J. 7, 308–313
- Shirazi, A., Ceberio, J., and Lozano, J. A., 2018. *Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions*, Progress in Aerospace Sciences, Vol. 102, pp. 76-98
- Url-1: GALib, A C++ Library of Genetic Algorithm Components, v. 2.4.7, lancet.mit.edu/ga, Access: 07.2019.
- Url-2: OptimLib, A lightweight C++ library of numerical optimization methods for onlinear functions, kthohr.com/optimlib.htm, Access: 09.2019.
- Valle, S. Member, G. K. Venayagamoorthy, S. Member, and R. G. Harley, 2008. *Particle swarm optimization: Basic concepts, variants and applications in power systems*, IEEE Trans. Evol. Comput. 12, 171–195
- West, C., Montazeri, A., Monk, S. D., and Taylor, C. J., 2016. *A genetic algorithm approach for parameter optimization of a 7DOF robotic manipulator*, International Federation of Automatic Control (IFAC), Vol. 49, pp. 1261-1266