

## Evrimsel Algoritma ile Parçacık Sürü Algoritmasının Simülasyon Tabanlı Karşılaştırılması ve Elde Edilen Sonuçların Analizi

Serkan Dereli<sup>1</sup>, Raşit Köker<sup>2</sup>, İsmail Öylek<sup>3\*</sup> ve Metin Varan<sup>4</sup>

<sup>1,3</sup> Bilgisayar Teknolojileri, Sakarya Meslek Yüksekokulu, Sakarya Üniversitesi, Türkiye

<sup>2</sup> Elektrik-Elektronik Mühendisliği, Sakarya Üniversitesi, Türkiye

<sup>4</sup> Elektrik-Elektronik Mühendisliği, Sakarya Üniversitesi, Türkiye

\* (ioylek@sakarya.edu.tr)

**Özet** – Evrimsel algoritma ve parçacık sürü algoritması çok uzun zamandır ayrı ayrı çeşitli karmaşık problemlerin çözümünde yaygın olarak kullanılan sezgisel algoritmalar. Evrimsel algoritma canlıların çoğalmasını ve hayatta kalmak için en iyilerin seçilmesi tekniğine dayanırken, parçacık sürü algoritması kuş ve balık sürülerinin yiyecek arama davranışlarından esinlenilerek geliştirilmiştir. Her iki algoritmanın da üstün olduğu taraflar vardır. Parçacık sürü algoritması iyi sonuçlara daha iyi yaklaşırken, evrimsel algoritma ise iyilerin yeni nesillere aktarılması ve adaptasyon neticesinde daha kısa çalışma zamanına sahiptir. İşte bu çalışmada konu edilen algoritmaların sahip olduğu bu üstünlüklerin simülasyon tabanlı bir uygulama ile açıkça gösterilmektedir.

**Anahtar Kelimeler** – evrimsel algoritma, parçacık sürü algoritması, optimizasyon, meta sezgisel yöntem

**Abstract** - Evolutionary algorithm and particle swarm algorithm are heuristic algorithms that have been widely used to solve various complex problems for a very long time. While the evolutionary algorithm is based on the idea of increasing living things and choosing the best ones for survival, the particle swarm algorithm has been developed by inspiring food search behaviors of birds and fish. There are sides where both algorithms are superior. While the particle swarm algorithm better approximates the results, the evolutionary algorithm has a shorter working time due to the transfer of goodness to the next generation and adaptation. In this work, it is a study in which the advantages of the algorithms are clearly demonstrated by a simulation-based application.

**Keywords** – evolutionary algorithm, particle swarm algorithm, optimization, heuristic method

### I. GİRİŞ

Günümüzde veri yoğunluğunun artmasıyla beraber matematiksel işlemlerin zorlaştığı ve karmaşık hale geldiği, zor olarak ifade edilen problem sayısının her geçen gün yükseldiği görülmektedir. Bu zor problemlerin kısa sürede ve doğru bir şekilde çözümü büyük önem arz etmektedir. Dolayısıyla son dönemlerde yeni yöntemler geliştirilerek çözümü zor problemlerin ele alınması araştırmacılar açısından ilgi odağı haline gelmiştir. İnsan beyninin çalışma yapısından yola çıkılarak geliştirilen yapay sinir ağları bu tekniklerden bir tanesidir [1]. Benzer şekilde bilim insanları, canlıların doğaya ayak uydurarak en iyi şekilde hayatta kalabilme özelliğinden ilham alarak genetik algoritma (GA), evrimsel stratejiler (EA), diferansiyel evrim (DE) gibi algoritmaları geliştirmişlerdir[2]. Daha sonra hayvan sürülerinin beraber hareket ettiklerinde çok başarılı işler çıkardıklarına tanık olmuşlar ve böylece parçacık sürü optimizasyonu (PSO), yapay arı kolonisi (ABC), ateş böceği (FA) gibi sürü algoritmalarını geliştirmişlerdir[3]. Bu çalışmada farklı yapıdaki bir sürü algoritması ile evrimsel algoritmanın karşılaştırılarak sonuçları analiz edilmektedir.

El-Abd diferansiyel evrim (DE), genetik algoritma (GA), harmoni arama (HS) ve parçacık sürü optimizasyonu (PSO) algoritmalarının başarımını doğrusal olmayan kıyaslama fonksiyonlarıyla ortaya koymuştur[4]. Fadee ve Radzi yaptıkları çalışmada evrimsel algoritma kullanılarak

oluşturulan hibrit yenilenebilir enerji sistemleri üzerine kapsamlı bir araştırma gerçekleştirmişlerdir[5]. Zhou ve arkadaşları çok amaçlı evrimsel algoritmaların (MOEA) en son teknolojideki konumlarını araştıran bir çalışma yapmışlardır[6]. Zhang yapmış olduğu çalışmada evrimsel algoritma fikrine kuantum mekanizmasını ilave ederek yeni bir teknik geliştirmiş ve geliştirilen bu fikrin evrimsel algoritmayı ne kadar iyileştirdiğini ortaya koymuştur[7]. Lara ve arkadaşları yeni bir yerel arama stratejisi geliştirerek bu stratejinin hem tek başına kullanıldığında hem de evrimsel algoritma ile birlikte kullanıldığında sonuçlarını performansları iyi bilinen karşılaştırma fonksiyonlarıyla ortaya koymuşlardır[8]. Mukhopadhyay ve arkadaşları yapmış oldukları çalışmada veri madenciliğinde özellik seçme ve sınıflandırma işlemlerinin gerçekleştirilmesinde çok amaçlı evrimsel algoritmanın kullanım şekline yönelik kapsamlı ve karşılaştırmalı bir araştırma ortaya koymuşlardır[9]. Martens ve arkadaşları evrimsel algoritmayı kullanarak performans, güvenilirlik ve maliyet açısından yazılım mimarisi modellerini otomatik olarak geliştirmeyi başarmışlardır[10]. Crepinsek ve arkadaşları evrimsel algoritmada arama ve keşfedilen noktaların en iyi değerlerini elde etmek için kullanılması işleminin nasıl olduğuna dair bir araştırma ortaya koymuşlardır[11].

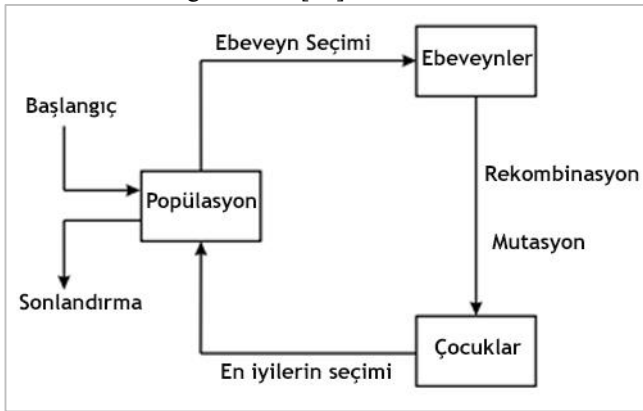
Dereli ve Köker yapmış oldukları çalışmada PSO algoritmasını kullanarak 7-dof seri robotun ters kinematik hesabını gerçekleştirmişler[12], Sharafi ve El-Mekawy ise

PSO kullanarak hibrit, yenilenebilir bir enerji sistemi geliştirmişlerdir[13]. Wang ve arkadaşları parçacık sürü optimizasyonu vasıtasıyla kristal yapılarının en iyi şekilde tahmin edilmesi üzerine bir çalışma gerçekleştirmişlerdir[14]. Bansal ve arkadaşları PSO algoritması ile elde edilen sonuçları daha iyi hale getirmek için IW stratejilerini çalışmalarında kullanmışlar ve karşılaştırmalı olarak sonuçları ortaya koymuşlardır[15].

## II. MATERYAL VE YÖNTEM

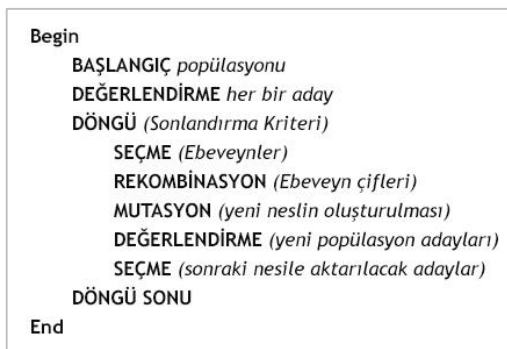
### A. Evrimsel Algoritma

Evrimsel Algoritma, 1960 yılından bu tarafa çok farklı şekilleri önerilmiş ve bilim insanları tarafından defalarca kullanılmıştır. Mevcut popülasyon içerisinde doğal seçim ve mutasyon neticesinde uyumlu bireylerin sonraki nesillere aktarılması mantığından yola çıkılarak bilim camiasına kazandırılan bir algoritmadır[16].



Şekil 1. Evrimsel algoritma akış şeması

Şekil 1'de evrimsel algoritmanın çalışmasına dair akış şeması görülmektedir. Popülasyon, muhtemel çözümleri tutmaktadır ve sürekli olarak en iyi nesillerle güncellenmektedir. Ayrıca popülasyonun çeşitliliği farklı çözümlerin sayısının bir ölçümüdür.



Şekil 2. Evrimsel algoritma

Ebeveyn seçme mekanizması iki önemli görevi yerine getirmektedir. Bunlardan ilki popülasyondaki iyi bireyleri seçmek, ikincisi ise seçilen bu iyi bireylerin sonraki nesilde ebeveyn olmasını sağlamaktadır. Öncelikle en iyi ebeveynler seçilmekte daha sonra da bu ebeveynlerin mutasyonu ile ilgili değerler sonraki nesillere aktarılmaktadır[17].

Evrimsel algoritmada mutasyon ve rekombinasyon olmak üzere iki adet varyasyon parametresi bulunmaktadır. Mutasyon tek bir bireyde meydana gelebilecek kalıcı değişimler olarak tanımlanırken rekombinasyon ise iki ebeveynin birleşmesi neticesinde meydana gelen değişimdir.

En iyilerin seçim mekanizması da tıpkı en iyi ebeveyn seçimi gibi bireyler arasındaki kaliteli olanların ayrılması işlemidir. Tek fark bu işlem evrimsel döngünün farklı bir sahnesidir. Uygunluk fonksiyonun devreye girdiği bölüm burasıdır.

### B. Parçacık Sürü Algoritması

İlk olarak Kennedy ve Eberhart tarafından optimizasyon tekniği olarak kullanılan Parçacık Sürü Algoritması (PSO), kuş veya balık sürülerinin yiyecek arama davranışlarından ilham alınarak üretilen güçlü bir arama algoritmasıdır[18]. Diğer meta-sezgisel algoritmalarla karşılaştırıldığında kontrol parametrelerinin sayısının az olması sebebiyle uygulaması kolay olan bir tekniktir[19].

Biyolojik sistemlerden esinlenen birçok hesaplama tekniği mevcuttur. Örneğin yapay sinir ağları, insan beyninin basitleştirilmiş bir modelidir[20]. Genetik algoritmalar, biyolojideki evrimsel süreçten esinlenir[21]. PSO'da ise ele alınan konu biyolojik sistemlerin farklı bir türü olan sosyal sistemlerdir. Özellikle birbiriyle ve çevresiyle etkileşim içinde olan basit bireylerin birliktelik davranışları incelenmektedir. Bu kavram parçacık zekâsı olarak isimlendirilir[22].

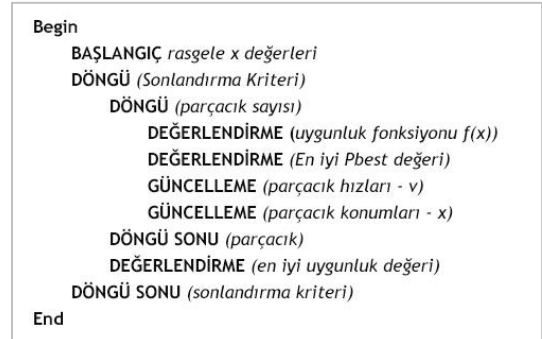
PSO, Genetik Algoritmalar gibi evrimsel hesaplama teknikleri ile birçok benzerlikler gösterir. Sistem rasgele çözümlerden oluşan bir popülasyonla başlatılır ve en iyi çözüm için jenerasyonları güncelleyerek arama yapar. Buna karşın, evrimsel algoritmanın aksine PSO'da çaprazlama ve mutasyon gibi evrimsel operatörler yoktur. PSO'da parçacık denilen potansiyel çözümler, mevcut en iyi çözümleri takip ederek problem uzayında gezinirler.

Bu algoritma doğrusal olmayan problemlerde yoğun olarak kullanılmaktadır. Bu problemler belli bir arama uzayına sahip önemli ve çözümlü zaman alan problemlerdir. PSO çözüme deneyim ve komşuluk ilişkilerini birlikte kullanarak ulaşabilmektedir.

$$v_{id} = v_{id} + c_1 \cdot r_1 \cdot (p_{best} - x_{id}) + c_2 \cdot r_2 \cdot (g_{best} - x_{id}) \quad (1)$$

$$x_{id} = x_{id} + v_{id} \quad (2)$$

Hız (1) ve konum (2) denklemleri olmak üzere sadece iki eşitliğe sahip olması nedeniyle uygulaması kolay olan bir algoritmadır. Bu iki eşitlik her iterasyonda güncellenir.



Şekil 3. Parçacık sürü optimizasyonu algoritması

Şekil 3'te görüldüğü gibi parçacık konumları (x) ve parçacık hızları (v) olmak üzere iki adet kontrol parametresi öne çıkmaktadır. Parçacıkların konumu her bir parçacığın hızına göre yeniden güncellenmektedir.

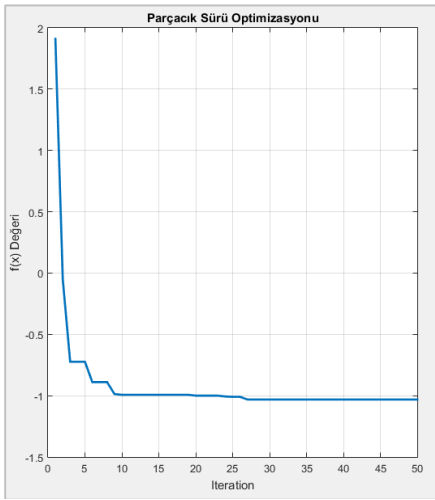
### C. Yöntem

Bu çalışmada mutasyon tabanlı evrimsel algoritma ile sürü tabanlı parçacık sürü algoritmasının simülasyon tabanlı karşılaştırılması gerçekleştirilmiştir. Karşılaştırma işlemi, sonuç değeri ve çözüm zamanı olmak üzere iki şekilde yapılmıştır.

$$f(x) = 4x_1^2 - 2x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4 \quad (3)$$

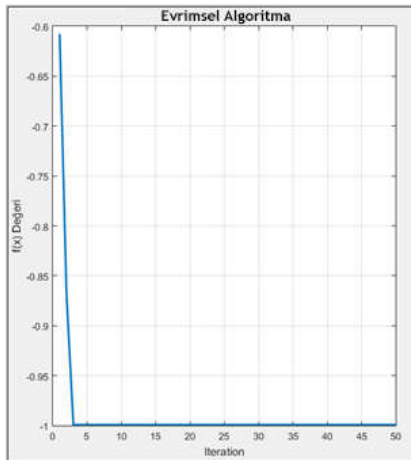
Her iki algoritma ile Six-Hump Camel (3) fonksiyonunun [-3, 4] değer aralığındaki en iyi değeri elde edilerek sonuçları analiz edilmiştir. Çalışma, Matlab ara yüzünde gerçekleştirilmiş olup her iki algoritmada da 10'ar defa çalıştırılarak bunlar arasından en iyi olanı seçilmiştir. Çalışmada algoritmalar ayrıca çalışma zamanı bakımından da karşılaştırılmıştır.

### III. BULGULAR



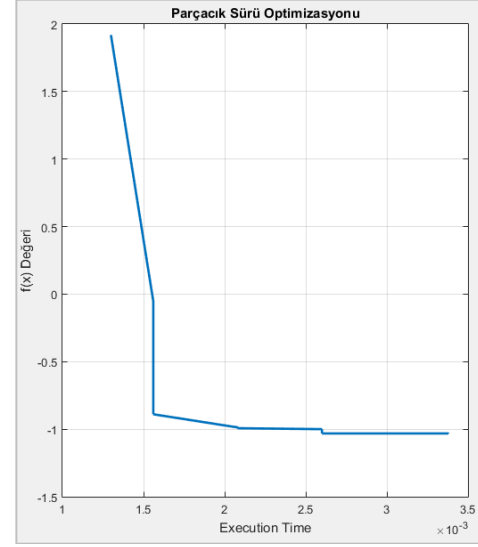
Şekil 4. PSO'nun sonuç grafiği

Şekil 4'te görüldüğü üzere parçacık sürü optimizasyonu 25. ile 30. iterasyonlar arasında doğru sonuca ulaşabilmiştir. Sonuca ulaşma hızı parçacık hızlarına bağlı olduğu için iterasyon sayısı artmıştır.



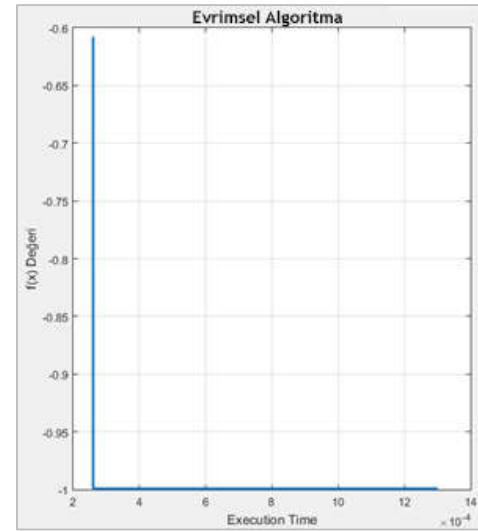
Şekil 5. Evrimsel algoritmanın sonuç grafiği

Şekil 5'te görüldüğü gibi evrimsel algoritma ise 5. ile 10. iterasyonlar arasında sonuca ulaşabilmiştir. Çünkü, bu algoritma en iyi değerlerin sonraki nesile aktarılması sayesinde, çözüme kısa sürede yani daha az iterasyon sayısında ulaşabilmektedir.



Şekil 6. PSO'nun çalışma süresi

Şekil 6'da görüldüğü üzere Parçacık Sürü Algoritması en iyi sonucuna 2.5 ms ile 3 ms arasında ulaşmıştır.



Şekil 7. Evrimsel Algoritmanın çalışma süresi

Şekil-7'te açıkça evrimsel algoritmanın çok daha kısa bir sürede çözüme ulaştığı görülmektedir. Çünkü PSO'da çözüm parçacıklarının hızlarına bağlıdır. Bu nedenle de çözüm zamanı uzun sürmektedir.

### IV. TARTIŞMA

Sonuçlardan da açıkça görüldüğü üzere parçacık sürü algoritmasında en iyi değer daha ileri iterasyon sayılarında elde edilirken; evrimsel algortmada daha düşük iterasyon sayılarında en iyi değer elde edilmiştir. Çünkü parçacıkların belli bir hızı bulunmakta ve hedefe bu hızlarla ulaşabilmektedirler. Bu durum da çözüm iterasyon sayısının ve süresinin artmasına sebep olmaktadır. Oysa evrimsel algoritma en iyi bireylerden yeni bir nesil oluşturarak hedefe ilerlediğinden çok kısa sürede sonuca ulaşmıştır.

### V. SONUÇ

Evrimsel algortmada ebeveyn sayısı 10, parçacık sürü algoritmasında ise sürü sayısı 10 olarak alınmıştır. Her iki algortmada da iterasyon sayısı 50 olarak belirlenmiştir.

Parçacık Sürü Algoritması sonuca en yakın değeri elde etmesine rağmen, Evrimsel Algoritma çalışma zamanı bakımından PSO algoritmasıyla karşılaştırıldığında 10 kata kadar daha kısa bir sürede, sonuca çok daha hızlı yaklaşmıştır. Başka bir deyişle Parçacık Sürü Algoritması iyi sonuçlara daha iyi yaklaşırken, Evrimsel Algoritma ise iyi bireylerin yeni nesillere aktarılması ve adaptasyon neticesinde daha kısa sürede sonuca ulaşma özelliğine sahiptir.

#### REFERANSLAR

- [1] S. Dereli, R. Köker, "Design and Analysis of Multi-Layer Artificial Neural Network Used for Training in Inverse Kinematic Solution of 7-DOF Serial Robot," *Gaziosmanpaşa Journal of Scientific Research*, vol. 6, pp. 60-71, 2017.
- [2] A.P.Engelbrecht. *Computational Intelligence*. London: Wiley, 2007.
- [3] Yang, X.S. *Engineering Optimization*. New Jersey: Wiley, 2010.
- [4] M.El-Abd, "Performance assessment of foraging algorithms vs. evolutionary algorithms," *Information Sciences*, vol. 182, pp. 243-263, 2010.
- [5] M.Fadee, M.A.M.Radzi, "Multi-objective optimization of a stand-alone hybrid renewable energy system: A review," *Renewable and Sustainable Energy Reviews*, vol. 16, pp. 3364-3369, 2012.
- [6] A.Zhou, B.Y.Qu, P.N.Suganthan, Q.Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, pp. 32-49, 2011.
- [7] G.Zhang, "Quantum-inspired evolutionary algorithms: a survey and empirical study," *Journal of Heuristics*, vol. 17, pp. 303-351, 2011.
- [8] A.Lara, G.Sanchez, C.A.Coello, O.Schütze, "HCS: A New Local Search Strategy for Memetic Multiobjective Evolutionary Algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 14, pp. 112-132, 2010.
- [9] A.Mukhopadhyay, U.Maulik, S.Bandyopadhyay, C.A.Coello, "A Survey of Multiobjective Evolutionary Algorithms for Data Mining: Part I," *IEE Transactions On Evolutionary Computation*, vol. 18, pp. 4-19, 2014.
- [10] A.Martens, H.Koziolok, S.Becker, R.Reussner, "Automatically Improve Software Architecture Models for Performance, Reliability, and Cost Using Evolutionary Algorithms," in *Proc International Conference on Performance Engineering*, 2010.
- [11] M.Crepinsek, S.H.Liu, M.Mernik, "Exploration and Exploitation in Evolutionary Algorithms: A Survey," *ACM Computing Surveys*, vol. 45, pp. 1-27, 2013.
- [12] S.Dereli, R.Köker, "IW-PSO approach to the inverse kinematics problem solution of a 7-Dof serial robot manipulator," *Sigma Journal of Engineering and Natural Sciences*, vol. 35, pp. , 2017.
- [13] M.Sharafia, T.Y.ELMekkawy, "Multi-objective optimal design of hybrid renewable energy systems using PSO-simulation based approach," *Renewable Energy*, vol. 68, pp. 67-79, 2014.
- [14] Y.Wang, J.Lv, L.Zhu, Y.Ma, "Crystal Structure Prediction via Particle Swarm Optimization," *Physical Review*, 2010.
- [15] J.C.Bansal, P.K.Singh, M.Saraswat, A.Verma, S.S.Jadon, A.Abraham, "Inertia Weight Strategies in Particle Swarm Optimization," in *Proc. IEEE Nature and Biologically Inspired Computing (NaBIC)*, 2011.
- [16] D.Lim, Y.Jin, Y.S.Ong, B.Sendhoff, "Generalizing Surrogate-Assisted Evolutionary Computation," *IEEE Transactions on Evolutionary Computation*, vol. 14, pp. 329-355, 2009.
- [17] A.Kusiak, H.Zheng, "Optimization of wind turbine energy and power factor with an evolutionary computation algorithm," *Energy*, vol. 35, pp. 1324-1332, 2010.
- [18] J.Kennedy, R.C.Eberhart, "Particle Swarm Optimization," in *Proc. IEEE international conference on neural network*, 1995.
- [19] R.Poli, J.Kennedy, T.Blackwell, "Particle Swarm Optimization," *Swarm Intelligence*, vol. 1, pp. 33-57, 2007.
- [20] S.S.Chiddarwar, N.R.Babu, "Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6R serial robot by a fusion approach," *Engineering Applications of Artificial Intelligence*, vol. 23, pp. 1083-1092, 2010.
- [21] R.Köker, "A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization," *Information Science*, vol. 222, pp. 528-543, 2013.
- [22] T.X.Yen, D.Sharma, D.Sirinivasan, P.N.Manji, "A modified particle swarm optimization approach for unit commitment," in *Proc IEEE Evolutionary Computation (CEC)*, 2011.