**2. Yöntem**

**2.1. Optimizasyon ile ilgili genel formüller**

**2.2. FDB-Selection Method**

**2.3. EFO**

**2.4. FDB-based EFO**

1. /\*initialization of a random electromagnetic field\*/
2. **for** i=1 to N\_emp **do**
3. **for** j=1 to N\_var **do**
4. em\_pop[i,j] = min + rand(0,1) \* (max - min)
5. new\_emp[j] = em\_pop[i,j]
6. **end for**
7. fit[i] = (new\_emp)
8. **end for**
9. sortPopulation(em\_pop, fit) **/\*sort population from best to worst based on fitness\*/**
10. **/\*main loop (optimization)\*/**
11. RI = 1
12. **while** (stop criterion is not meet) **do**
13. force = rand(0,1)

Selection Process (same as original EFO)

1. **for** i=1 to N\_var **do**
2. I\_pos = rand(1, floor(N\_emp \* P\_field))
3. I\_neu = rand(ceil(N\_emp \* P\_field), ceil(1 – N\_field) \* N\_emp)

Selection Process (proposed in FDB-based EFO)

1. **if (rand(0,1) < N\_var / (100\*5 ))**
2. **I\_neg = fdb(N\_emp, fit)**
3. **else**
4. **I\_neg = rand(floor(1 – N\_field) \* N\_emp), N\_emp)**
5. **end if**

Search process (exploitation phase)

1. **if** (rand(0,1) < Ps\_rate) **then**
2. new\_emp[i] = emp\_pop[I\_pos, i]
3. **else**
4. new\_emp[i] = emp\_pop[I\_neu, i] + phi \* force \* (emp\_pop[I\_pos, i]
5. – em\_pop[I\_neu,i]) – force \* (em\_pop[I\_neg, i] – em\_pop[I\_neu, i])
6. **end if**

Bound control process

1. **if** (new\_emp[i] > max pr new\_emp[i] < min) **then**
2. new\_emp[i] = min + rand(0,1) \* (max - min)
3. **end if**
4. **end for**

Search process(exploration phase)

1. **if** (rand(0,1) < R\_rate) **then**
2. new\_emp[RI] = min + rand(0,1) \* (max - min)
3. RI++
4. **if** (RI > N\_var) **then**
5. RI = 1
6. **end if**
7. **end if**
8. New\_fit = f(new\_emp)
9. **if** (New\_fit < worst(fit) ) **then**
10. insertInSortedPopulation(new\_emp)
11. **end if**
12. **end while**
13. **return** em\_pop[1,1:N\_var]

**Fig. X.** General flow of FDB-based EFO

**Eklenen kodun açıklaması:** EFO algoritmasının baz modelinde çözüm adayları topluluğu (elektromnanyetik parçacıklar) rasgele yaratıldıktan sonra üç alt popülasyona bölünmektedir. Bunlar doğada olduğu gibi pozitif, nötr ve negatif yüklü parçacıkları temsil etmektedirler. Alt popülasyonlardaki parçacık sayıları daha önceden belirlenen oranlara göre ve parçacıkların uygunluk değerlerine bağlı olarak ayarlanmaktadır. Buna göre parçacıklar uygunluk değerlerine bağlı olarak sıralanmakta ve aç gözlü seçim yöntemi ile alt popülasyonlara dağıtılmaktadırlar. EFO algoritmasında, uygunluk değerlerine bağlı olarak sıralanan popülasyonda ilk 10%’da yer alan parçacıklar seçilerek pozitif alanı temsil eden alt popülasyon oluşturulur. Sıralamanın 10%-55% arası dilimde yer alan çözüm adayları nötr alanı ve son 45%’lik dilimde yer alan parçacıklar ise negatif alanı temsil eden alt popülasyonları oluşturmaktadırlar. Alt popülasyonları oluşturduktan sonra arama süreci yaşam döngüsü başlamaktadır. Bu süreçte ilk olarak elektro mıknatıslar arasındaki itme-çekme kuvvetini temsil eden parametre değeri force=rand (0,1) ifadesi ile hesaplanır. Arama süreci yaşam döngüsü temel olarak üç adımdan oluşmaktadır. Bunlar; seçim işlemleri (arama sürecine rehberlik eden parçacıkların belirlenmesi), komşuluk araması ve çeşitlilik adımlarıdır. Seçim işlemleri evresinde EFO popülasyonunu oluşturan üç alandan birer adet parçacık rasgele bir şekilde seçilir. Bunlar pozitif, nötr ve negatif alanlardan rasgele seçilen çözüm adaylarıdır. Önerilen yöntemde de pozitif ve nötr alanlardan parçacık seçim işlemi klasik EFO algoritması ile aynıdır (Şekil X’de 15 ve 16. satırlar). Bununla birlikte, önerilen yöntemde negatif alandan parçacık seçim işleminde FDB yöntemi de kullanılmaktadır. FDB yönteminin tatbik edilmesine ilişkin sözde-kod Şekil X’deki 17-21 arası satırlarda verilmektedir. Bu satırlardaki sözde kodlardan da anlaşılacağı üzere negatif alandan seçim işlemi problem boyutuna (N\_var) bağlı olarak olasılıksal olarak işleyen iki alt adıma bölünmüştür. Problem boyutunun 20%’sine karşılık gelen bu olasılık “rand(0,1) < N\_var / (100\*5 )” formülü ile hesaplanmaktadır. Buna göre problem boyutunun 20%’si oranında bir sıklıkla FDB yöntemi kullanılarak popülasyondaki tüm çözüm adayları arasından bir adet parçacık seçilmektedir. Problem boyutunun 80%’ine karşılık gelen oranda ise klasik EFO algoritmasında olduğu gibi sadece negatif parçacıklar arasından ve rasgele bir şekilde seçim yapılmaktadır. Seçim işleminden sonra arama sürecinin diğer iki önemli evresi olan komşuluk araması ve çeşitlilik işlemleri gerçekleştirilmektedir. Komşuluk araması görevi, Şekil X’de 22-27 satırları arasında verilen işlemlerle ve iki farklı matematiksel ifade ile gerçekleştirilmektedir. Komşuluk araması sürecinde kullanılacak olan yöntemin belirlenmesinde Ps\_rate yani pozitif alandan seçilme oranı kullanılmaktadır. Bu oran EFO algoritmasında 20% olarak kabul edilmiştir. Pozitif alandan seçilen çözüm adayının komşuluğunda arama yapmak için Şekil X’de 23. satırda verilen eşitlik kullanılmaktadır. pozitif, negatif ve nötr alanlardan seçilen çözüm adaylarının komşuluğunda arama yapmak için ise Şekil X’de 25. Satırda verilen eşitlik kullanılmaktadır. Arama süreci yaşam döngüsünün son adımı keşif (exploration) sürecidir. Bu sürece ait işlemler Şekil X’de 32-34 satırlarında gerçekleştirilmektedir. Bu satırlarda yer alan RI, üzerinde değişiklik yapılan tasarım parametresini temsil etmektedir. bir başka ifadeyle çözüm adayının (parçacığın) mutasyona uğrayacak olan boyutu RI ile gösterilmektedir. Bu süreçte üretilen çözüm adayının (new\_emp) uygunluk değeri ile popülasyondaki en düşük uygunluk değerine sahip olan çözüm adayı karşılaştırması yapılarak popülasyon güncellenir (Şekil X’de 35-42 satırlar).

Not