|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Nedir | TARİH : 11.06.2019 |
| HAVELSAN 30 yıldır Komuta Kontrol, Simülasyon, Eğitim ve Bilgi Sistemlerini Türk Silahlı Kuvvetleri ve tüm dünyada kullanılmak üzere geliştirmektedir. DOOB ürün ailesi, HAVELSAN’ın engin saha ve mühendislik tecrübesinin sonucunda geliştirilmiş ürünlerdir. DOOB ürün ailesi, Komuta ve Kontrol Bilgi Sistemlerinin temel niteliklerini içeren ve stratejik, operasyonel ve taktik komuta seviyelerinde kullanılmak üzere tasarlanmış üç farklı uygulamadan oluşur. DOOB uygulamalarının temel amacı tüm komuta seviyelerindeki durumsal farkındalığı arttırmaktır.  DOOB uygulaması farklı uluslararası ve milli standartlar vasıtasıyla farklı kaynaklardan toplandığı bilgileri entegre edip, Cari Harekât Takibi Modülü aracılığıyla bu bilgileri birleştirilmiş bir arayüzle sunarak durumsal farkındalığı arttırır. Bunun yanında, farklı fonksiyonel alanlarda hiyerarşik ve eş zamanlı planlama ve bu planların icra safhalarının farklı fonksiyon alanlarına uygun olarak takibini de destekler. DOOB milli ve NATO ihtiyaçlarını desteklediği gibi özel gereksinimlere kolayca uyarlanabilir ve genişletilebilir. DOOB ürün ailesi, HAVELSAN tarafından geliştirilmiş bir CBS çözümü olan TMAP ile desteklenmektedir.  DOOB web tabanlı bir uygulamadır ve modern web tarayıcısı haricinde, istemci makinesine herhangi bir kurulum gerektirmeden çalıştırılabilir. Kullanım uzmanları, DOOB ürünlerinin tasarım ve geliştirilmesi safhalarında yer aldıkları için her üç ürünün de kullanımı modern web uygulamalarına benzer, kolay ve hızlıdır. DOOB modüler bir uygulamadır ve farklı modülleri ayrı olarak kurulabilir. Bu nedenle kurulumu, bakımı ve yönetimi kolaydır.  DOOB Tactical uygulaması komuta yerlerinde taktik seviye komutanlar tarafından kullanılmak üzere geliştirilmiştir. DOOB Tactical ile taktik kullanıcılar DOOB tarafından üretilen planların basit halini alabilirler. Bununla birlikte kendi planlarını yapabilir ve cari harekatı takip edebilirler. Taktik kullanıcılar semboloji standartlarına uygun olarak genel durumsal farkındalığa bilgi girişi yaparak katkı sağlayabilir ve bu bilgiyi DOOB ve DOOB Mobile ile paylaşabilirler. DOOB Tactical, DOOB Mobile ile entegre olduğu için kendi birliğine ait DOOB Mobile kullanıcılarını izleyebilir.  SmartOps olarak da bilinen DOOB Mobile uygulaması, erler için geliştirilmiş ve bu seviyedeki komuta kontrol faaliyetlerini gerçekleştirmeyi hedefleyen taktik seviye bir uygulamadır. Android işletim sistemi tabanlı mobil uygulamadır. Güvenli VPN üzerinden temel olarak konum ve kalp atışı bilgilerini paylaşır, kendi takımındaki diğer erlerin aynı şekilde durumlarını takip eder ve bağlı olduğu DOOB Tactical komutanının konumunu görebilir. Bunun yanında canlı video yayımı, mesajlaşma, ses, fotoğraf ve diğer doküman paylaşım özellikleri de mevcuttur.  DOOB ürün ailesi kendi arasında entegre bir şekilde çalışır. Bu yüzden DOOB Tactical ve DOOB Mobile tarafından yayınlanan bütün bilgiler DOOB tarafından izlenebilir.  DOOB web tabanlı ve dağıtık mimariye sahip bir uygulama olduğu için tek bir sunucu üzerinden birçok kullanıcıya destek verebileceği gibi dağıtık mimari sayesinde bütün bir orduyu destekleyebilir. Bu şekilde kolay bir şekilde kurulup, kolay bir şekilde yönetilebilir ve genişletilebilir. DOOB ürününün geliştirilmesinde modern ve açık kaynak kodlu web ve uygulama çatıları kullanılmış ve herhangi bir şekilde rahat veya paralı bir kütüphane kullanılmamıştır. DOOB kullanıcının aşina olduğu diğer web uygulamalarına yakın bir kullanım kolaylığı sağlar. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 12.06.2019 |
| DOOB Tactical uygulaması komuta yerlerinde taktik seviye komutanlar tarafından kullanılmak üzere geliştirilmiştir. DOOB Tactical ile taktik kullanıcılar DOOB tarafından üretilen planların basit halini alabilirler. Bununla birlikte kendi planlarını yapabilir ve cari harekatı takip edebilirler. Taktik kullanıcılar semboloji standartlarına uygun olarak genel durumsal farkındalığa bilgi girişi yaparak katkı sağlayabilir ve bu bilgiyi DOOB ve DOOB Mobile ile paylaşabilirler. DOOB Tactical, DOOB Mobile ile entegre olduğu için kendi birliğine ait DOOB Mobile kullanıcılarını izleyebilir.  Askeri birimler tablete kurulan yazılım ile haberleşmektedir. Uygulama açıldığında karşınıza dünya haritası gelmektedir. Katagorilere ayrılmış şekiller ile harita üzerinde düşman birimleri, kendinize birimlerinizi ve harakat alanları çizilebilmektedir. Aynı internet ortamına bağlı cihazlar birbirlerini eklemektedirler. Açılan rapor türlerine göre real time olarak o anda açık olan tabletlere bilgiler göndermektedir. Verilerin gönderiminde Nato standartı olan VMF kullanılmaktadır.  Proje hakkında yeterli sözel bilgiyi edinip gün sonuna doğru kod incelemesine başlanmıştır. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 13.06.2019 |
| Projenin kodlarını inceleme devam ederken projenin geliştirildiği ide hakkında araştırma yapmaya başlandu. Proje Intelij Ideada geliştirilmekte. JetBrains isimli bir şirketin geliştirdiği ve sattığı bir idedir. Java ve Java ile yazılmış pekçok frameworke desteği vardır ve Ideada java ile yazılmıştır. Java dışında HTML5, JavaScript, PHP, Ruby, Python gibi farklı dilleri de desteklemektedir. Bazı kullanışlı kısa yol tuşları ise:   * Ctrl + n = Projede ismi girilen class veya interfaceleri arar * Ctrl + f12 = Sınıftaki metotları ve fieldleri gösterir * Ctrl + o = Override edilebilecek metotlar * Ctrl + h = Sınıf veya interface kalıtım hiyerarşisini gösterir. * Ctrl + shift + f = Projede girilen string ile eşleşenleri bulur * Ctrl + q = Seçilen bir fonksiyon veya sınıf hakkında bilgi verir * Shift + f6 = Seçilen bir metot veya fieldin isimi yeniden adlandırır. Adlandırma işlemi o sayfada kullanıldığı tüm yerler için geçerlidir. * Alt + sol / sağ yön tuşları = Açılan sayfalar arasında geçiş işlemi için kullanılır. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 14.06.2019 |
| DOOB tactical projesinde proje yönetim aracı olarak maven kullanulmaktadır. Maven, ilk versiyonu 30 Mart 2002’de çıkan proje bağımlılık yönetim aracıdır. Hatta direk kendisine “Proje Yöneticisi” denilebilir. Ant’a bir alternatif olarak gösterilen aynı Ant gibi bir JDT(Java Development Tool) ‘dur. Bağımlılık yönetiminin yanında modüler bazda proje geliştirme imkanı, plugin desteğiyle proje yönetimini daha güçlü kılma ve benzeri imkanlar da sunar.  Maven, proje geliştirirken proje içerisinde bir standart oluşturmamızı, geliştirme sürecini basitleştirmemizi, dokümantasyonumuzu etkili bir şekilde oluşturmamızı, projemizdeki kütüphane bağımlılığını ve IDE bağımlılığını ortadan kaldırmamızı sağlayan bir araçtır. Bizi internette sayfa sayfa dolaşıp ihtiyacımız olan library(kütüphane)’nin ihtiyacımız olan versiyonunu arama zahmetinden kurtaran bir çatıdır. Yani projeniz örneğin log4j.jar ve Spring.jar gibi kütüphanelere ihtiyaç duyuyorsa; bu ihtiyacı ilgili jar dosyalarını bularak ilgili yerlere kopyalamak mavenin sorumluluğundadır. Pom.xml (maven in proje ayar dosyası) içerisinde log4j nin şu versiyonu ve Spring’ın şu versiyonu kullanmak istiyorum demeniz yeterlidir. Kullanabilmemiz için çeşit çeşit konfigürasyonlar yapmamıza gerek yoktur. Böylece Projelerinizi istediğiniz yere kolayca taşımanıza + kütüphanelerin yönetilmesine imkân sağlar.  Örneğin; Spring Core kütüphanesini projemize dahil edebilmek için aşağıdaki XML kodunu pom.xml dosyamıza eklememiz gerekiyor.  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-core</artifactId>  <version>4.3.2.RELEASE</version>  </dependency>  Kısaca etiketlerin özelliklerine değinmek gerekirse;   * **<dependency>** : Bir bağımlılık temsil eder. İçerisinde bağımlılığa ait özellikleri barındırır. * **<groupId>** : Bağımlılığın hangi organizasyona ait olduğunu ifade eder. Burada org.springframework aslında birçok kütüphane barındıran bir organizasyondur. Java dünyasında groupId genellikle projenin web adresi ile alakalıdır. Eğer projenizin web adresi “blogprojem.com” ise groupId “com.blogprojem” olabilir. Bu bir şart değil, sadece genel bir yazılı olmayan kuraldır. * **<artifactId>** : Belirtilen organizasyon içinde hangi kütüphanenin getirileceğini ifade eder. * **<version>** : Anlaşılacağı üzere hangi versiyonun getirileceğini ifade eder. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 17.06.2019 |
| Projedeki gereksinimlerden birisi uygulamanın yüklendiği cihazlar birbirlerini realtime olarak takip etmesidir. Realtime haberleşme için lan ortamı kullanılacaktır. İnternet üzerinde bulunan bir server/websocket yapısı kullanılmayacağından iki farklı çözüm bulundu. Bu çözümler:   * Ortak lan’a bağlı cihazların ip adresleri yazılım aracılığı ile veri tabanına kaydedilerek, anlık iletişim sırasında veri tabanına eklenen ip adreslerine mesaj göndermek * Her cihaz ortak lan’a belli aralıklarla broadcast mesajı atıcak. Mesaj iceriği ip adresi veya urn olabilir. Mesajı alan cihaz kendi tablosunda bilgisi bulunmayan cihaz tespit eder geri bildirim yaparak noktadan noktaya bilgi alışverişi ile cihazdan bilgilerini ister. Anlık iletişim sırasında ise veri tabanına eklenen ip adreslerine bilgiler gönderilir.   Bulunan çözümlerden 2.si seçilerek gerekli araştırmalara başlandı. Kullanılacak cihazların UDP desteğinden dolayı UDP temelli veri alış verişini sağlayacak yapılar bulunarak çalışma günü bitirildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 18.06.2019 |
| Broadcast mesajı bire bütün bir iletişim türüdür. Amaç oluşturulan paketi ağdaki tüm düğümlere göndermektir. Noktadan noktaya iletişimin aksine, hedef ana bilgisayarın ip adresini bilmek zorunda değiliz. Bunun yerine, bir yayın adresi kullanılır. IpV4 protokolü uyarınca, bir yayın adresi, ağa bağlı cihazların paketleri almak için etkinleştirildiği mantıksal bir adrestir. Örneği verilecek kod resminde yerel ağın yayın adresi olan 255.255.255.255 numaralı belirli bir ip adresini kullanırız.  İletişim tekniği için UDP terçih edilmişti. Java’da java.net paketi, UDP protokolü aracılığıyla iletişim kurmak için kullanılabilecek DatagramPacket ve DatagramSocket sınıflarını sunmaktadır. UDP tipik olarak ses/video akışı, ağ keşfi vb. gibi düşük gecikmenin garantili teslimattan daha önemli olduğu senaryolarda kullanılmaktadır. Datagram ise ağ üzerinden, varış, varış zamanı ve içeriği garanti edilmeyen, bağımsız ve kendi kendine yeten bir mesajdır.  Aşağıdaki örnek kodda UDP protokolü kullanılarak oluşturulan paketin ağdaki tüm düğümlere göndermek hedeflenmiştir. 4445 değeri datagramın göderildiği uzak ana bilgisayardaki port numarasıdir. Diğer bir deyişle sanal veri yolu numarasıdır.  Aşağıdaki örnek kodda ise ağ ortamından gönderilen paketleri almak hedeflenmiştir.  Projede verilen kod örneklerinden yararlanılarak UDP kullanılarak veri gönderme modülü eklendi. Veri alma ve test işlemleri yarına bırakıldı. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 19.06.2019 |
| Projeye UDP veri alma modülü eklendi. Veri alma ve gönderme işlevleri anlık olması için thread yapısı kullanılmıştır. Oluşturulan senaryo ise: Uygulama açıldığı anda her 14sn bir cihaz kendi ipsini lana bırakır. Dinlemede olan cihazların, lana bıraklan ıp adresini alarak kendilerinde olup olmadığını kontrol ederler. Kendisinde olmayan ip’li cihazın entity bilgisini almak için ilgili cihaza istek mesajı gönderilir. İstek mesajını alan cihaz, entity bilgisini isteği gönderen cihaza gönderir.  Tabletlere yüklenilen uygulama çeşitli ayarlarda test edildi. Veri almada problemler çıktı. Veri gönderen cihaz sadece kendi gönderdiği verileri almaktaydı. Gerekli notlar alınarak 7. iş günü bitirildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 20.06.2019 |
| Tüm gün veri alma ve gönderme modüllerinde değişiklikler ve testler yapıldı. Genel sorun, sisteme girip cikan tabletin kendi entity’lerini diğer tabletlere göndermeme çözülemedi. Havelsana özel ağda çalışıldığından, bir engelleme olabileceği fikri paylaşıldı. Telefon internetine bağlanan cihazlarda aynı sorun devam etti. Gün sonunda karşılaşılan problemlere çözüm bulunamadı. Broadcast yerine multicast yapısı kullanılmaya karar verildi. Gerekli araştırmalara başlanıp 8. iş günü bitirildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 21.06.2019 |
| Optimizasyon alanında gerçek dünya problemleri karmaşık ve çözümü zor olan problemlerdir. Bu tür karmaşık ve çözümü zor olan problemlerde genellikle tam sonuç elde etmek için kullanılan algoritmalar yürütme zamanı açısından yavaş ve sadece çözmek için tasarlandığı problemde işleyecek biçimde tasarlanırlar. Bu tür algoritmaları başka zor ve karmaşık problemlerde kullanmak imkânsızdır. Bu sebeple kesin çözümü vermeyen ve daha hızlı çalışan algoritmalar geliştirilmiştir. Bu algoritmalara sezgisel algoritmalar denir.  Sezgisel algoritmalar diğer algoritmalara göre daha hızlıdır ve çözüm uzayındaki bütün olasılıkları değerlendirerek en iyi çözüme yakın bir çözümü bulmayı amaçlarlar. Ancak en iyi çözümün bulunacağını hiçbir zaman garanti etmezler. Bu tür algoritmalar hızlı çalışmalarına rağmen geliştirilme aşamasında üzerinde kullanılacağı problemin bilgilerinden yararlanıldığı için uygulanıldığı probleme bağlı olan algoritmalardır ve bu algoritmalar klasik sezgisel algoritmalar olarak adlandırılır. Klasik sezgisel algoritmalara A\* Araması, Demet Araması, Tırmanış Araması, En İyi Öncelikli Arama, Açgözlü En İyi Öncelikli Arama algoritmaları örnek olarak verilebilir.  Klasik sezgisel algoritmaların dışında probleme bağlı olmayan meta sezgisel algoritmalar vardır. Meta kelime anlamı olarak üst seviye anlamına gelir. Yani meta sezgisel algoritmaları, üst seviye sezgisel algoritmalar olarak değerlendirebiliriz. Bu algoritmalar genellikle doğadan ilham alınarak tasarlanmışlardır ve birçok probleme sadece amaç fonksiyonunu değiştirerek uygulanabilir. Meta sezgisel algoritmalar problem hakkında hiçbir bilgiye sahip olmadığı halde en uygun çözüm için en uygun değişken değerlerini verebildiği için bir nevi kara kutu olarak düşünülebilir. Meta sezgisel algoritmalara Genetik Algoritma, Karınca Kolonisi Optimizasyonu, Yapay Arı Kolonisi Optimizasyonu, Parçacık Sürü Optimizasyonu örnek olarak verilebilir.  Meta sezgisel algoritmalara verilecek değerli örneklerden biri de Yapay Arı Koloni Algoritması’dır. Yapay Arı Kolonisi Algoritması, Derviş Karaboğa tarafından 2005’te arıların topluluk olarak yiyecek arama davranışlarını temel alarak geliştirilmiş bir optimizasyon algoritmasıdır. Bu algoritma, bal arısı sürülerinin kendilerine özgü zeki davranışlarını örnek alarak, arıların besin ararken kullandıkları yöntemlerden esinlenerek oluşturulmuş bir optimizasyon algoritmasıdır. Sosyal bir düzen içinde yaşayan bal arıları, içgüdüsel olarak bu düzeni bilir ve hayatını bu kurallara uygun olarak devam ettirir. Kovandaki her arının görevi bellidir ve arılar asla bu görevlerinden sapmazlar. Belirtilen sosyal düzen, yiyeceklerin depolanması, balın getirilmesi, iletişim ve besin arama gibi işleri kapsamaktadır. Koloni halinde yaşayan bu sosyal yaşamda 3 arı çeşidi bulunmaktadır: Bunlar kraliçe arı, erkek arı ve dişi olan işçi arılardır. İşçi arılar, gözcü arılar ve kâşif arılar olmak üzere yapay arı kolonisi algoritmasında da üç grup arı bulunmaktadır. Bu modelde işçi arılar kaynaklara gönderilerek besin miktarlarını hesaplar. Aynı zamanda da gözcü arılar da başka kaynaklara gönderilerek besin miktarları hesaplanır. Daha sonra rasgele yeni kaynaklar bulmaları için kâşif arılar gönderilir. Durdurma kriteri sağlanana kadar bu döngü devam eder. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 24.06.2019 |
| Değişken Mesaj Biçimi (VMF) Askeri Standardı (MIL-STD), mesaj, veri öğesi ve protokol standartları dahil olmak üzere Ortak birlikte çalışabilirlik standartlarına sahip askeri hizmetler ve ajanslar sağlar. Bu standartlar, Ortak sınırlar arasında zamanında, kritik, komut ve kontrol bilgilerinin değiş tokuş edilmesini gerektiren otomatik taktik veri sistemlerinin (TDS) tasarımı, geliştirilmesi, test edilmesi, sertifikalandırılması, kullanılması ve sürdürülmesi için şarttır.  VMF tipik olarak bir kara kara kuvveti taktik veri bağlantısı (TDL) olarak düşünülürken, VMF istenen ortam ne olursa olsun destekleyecektir. Bu, özellikle, ABD, NATO ve Koalisyon güçleri arasında, kara, hava ve deniz iletişimini içeren bir savaş kaosunun içinde önemli bir durumsal değişim alışverişini sağlayan bir iletişim sistemi olan Link 16 TDL ile birlikte kullanıldığında önemlidir.  VMF, Combat Net Radio (CNR) ağlarının konuşlandırılması ve kullanılması durumunda iletişim bant genişliği kısıtlı durumları için çok uygundur. VMF ve Ortak Seri Genişletme Uygulamaları Protokolü (JREAP), hem dikey hem de yatay olarak çeşitli katılımcı düğüm, platform ve birimler ağına bağlanarak ses, görev ve taktik veri ve görüntü alışverişini mümkün kılar.  VMF yaygın olarak kullanılır, çünkü durum ne olursa olsun, UHF, VHF, HF ve SATCOM gibi çok çeşitli RF taşıyıcıları (radyo frekansı kanalları) üzerinde çalışabilme özelliğine sahiptir. Bir başka VMF iletişim özelliği: VMF sistemleri Ethernet ilkelerini, yerel alan ağlarında (LAN), büyükşehir alan ağlarında (MAN) ve geniş alan ağlarında (WAN) yaygın olarak kullanılan bir bilgisayar ağ teknolojisi ailesi kullanır.  Bir iletişim protokolü olarak VMF, bir iletişim sistemindeki iki veya daha fazla tarafın bilgi aktarmasına izin veren bir kurallar sistemini izler. Protokol kuralları sözdizimini, anlambilimi ve iletişimin senkronizasyonunu ve olası hata kurtarma yöntemlerini tanımlar. Protokoller donanım, yazılım veya her ikisinin bir kombinasyonu ile uygulanabilir. VMF, çoğu müttefik ülkenin özümsemeye çalıştığı J Serisi Taktik Veri Bağlantıları ailesinden biridir. J Serisi ailesinde Link 16, Link 22 ve IBS (Integrated Broadcast Service) bulunmaktadır. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 25.06.2019 |
| Broadcast yapısının yerine multicast yapısı kullanılmaya karar verildi.  Broadcast Haberleşmeyi almak isteyip istemediklerine bakılmaksızın, paketler ağdaki tüm düğümlere gönderildiği için yayın yetersizdir. Bu bir kaynak israfı olabilir.  Multicast bu sorunu çözer ve yalnızca ilgilenen tüketicilere paketler gönderir. Çok noktaya yayın, çok noktaya yayın adresinin her grubu temsil ettiği bir grup üyeliği kavramına dayanır.  IPv4'te, 224.0.0.0 ile 239.255.255.255 arasındaki herhangi bir adres çok noktaya yayın adresi olarak kullanılabilir. Yalnızca bir gruba abone olan düğümler, gruba iletilen paketleri alır.  Java'da, MulticastSocket, çok noktaya yayın IP'sine gönderilen paketleri almak için kullanılır. Aşağıdaki örnek MulticastSocket kullanımını gösterir:  Multicast kodu projeye eklenip test çalışmalarına başlandı. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 26.06.2019 |
| Bugün projenin ikonlarının yenilenme görevi alındı. Projenin görsel tarafı Java FX ile yazıldığından Java FX araştırması yapıldı.  Java FX, Java teknolojisiyle birleşerek masaüstü ve web uygulamaları gibi bir çok uygulama geliştirme olanağı sağlayan bir yazılım teknolojisidir. Görsel ve işitsel web öğeleri kullanarak zengin içerikli, etkileşimli ve kapsamlı uygulamalar geliştirmenizi sağlar. Uygulama geliştirme esnasında herhangi bir Java kitaplığını kullanma olanağının yanı sıra Java teknolojilerinin geliştirilmesi fırsatını tanır. Java FX ile tarayıcıda çalıştırılabilecek uygulamaları rahatlıkla geliştirebilir ve bunları masaüstüne taşıyabilirsiniz. Örnek vermek gerekirse Visual studio da geliştirilen “windows form application” uygulamalarının Java da ki karşılığı diyebiliriz.  Scene builder Java Fx ile birlikte gelen programlarımızda sürükle bırak mantığı ile çalışan görsel ara yüz hazırlamamızı sağlayan ileri yazılım teknolojisidir. Her bir elemente farklı id atanmasını sağlar. Sürükle bırak mantığı ile nesneler oluşturabiliriz (label, button, textview gibi). Nesnelere bir çok fonksiyon eklememizi sağlayan properties özelliği vardır ve hepsi hazırdır. Sadece istediğimiz özelliği tıklayarak ekleyebiliriz. Bütün arayüz kodları Fxml denilen kodlara çevrilerek geliştiricilere büyük kolaylık sağlar.  Aşağıdaki resim örneklerinde FXML yapısı ve JavaFx ile C# windows form application benzerliği görülmektedir. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 27.06.2019 |
| Harita üzerinde daha iyi durması için özel olarak tasarlanan iconlar tasarım ekibinden alındı. %80 %90 ve %100 saydamlıkta hazırlanmış iconları, projede yerleştilecek yerlerine göre yeniden adlandırmaya başlandı. Adlandırma işlemi bittiken sonra ImageView taglarının url kısımlarını değiştirmeye başlandı. Fxml uzantılı sayfalar sırasıyla %80, %90 ve %100 saydamlıktaki iconlar ile değiştirildi. Tablete atılan kodun görseli incelendi. İconların uygun olduğuna karar verildi. Her saydamlıktan 2 şer tane olacak şekilde tekrardan url kısımları değiştirildi. Takım liderimiz Levent Tanın’a sunum yapıldı ve %90 oranına karar verildi. Tüm iconlar %90’a ayarlanıp çalışma günü tamamlandı. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| CSA Algoritma kargaların zeki davranışlarını dikkate alarak geliştirilmiştir. Kargalar fazla yiyeceklerini bazı gizli noktalara saklarlar ve ihtiyaç duyduklarında kullanırlar. Kargalar daha iyi besin kaynağı bulmak için diğer kargaları takip ederler. Kendisinin takip edildiğini anlayabilen karga, onu takip eden kargayı şaşırtmak için başka konumlara uçabilmektedir. Böylece bir karganın gizli yiyecek deposunu bulması zorlaşır.  Optimisazyon bakış açısı ile kargalar, arayıcıları; çevre aramaya uzayını; çevredeki her posizyon uygun bir çözüm, besin kaynağının kalitesi, uygunluk fonksiyonunun ve en iyi besin kaynağı global çözümü ifade eder. Bu benzerlikler üzerinden CSA optimisazyon problemlerindeki en iyi çözümü bulmaya çalışmaktadır. CSA’nın dayandığı prensibler aşağıdaki gibidir;   * Kargalar sürü halinde yaşarlar * Kargalar yiyecek saklama yerlerinin posizyonlarını hatırlarlar. * Kargalar hırsızlık yapmak amacıyla birbirlerini takip ederler * Kargalar bir olasılık değerine göre yiyecek saklama yerlerini hırsızlığa karşı korurlar   CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Karga sürüsü rastgele biçimde üretilir   + Üretilen karga sürüsünün uygunluk değerleri bulunur ve hafızaya eklenir. * Takip aşaması – Konum Güncelleme   + Karga j, karga i’nin kendisini takip ettiğini bilmediği varsayılıyor. Bu durumda aşağıda verilen denklem ile karga i ‘nin konumu güncellenir. **X,** i kargasının konumunu; **r**, 0-1 arasında üretilmiş bir sayıyı; **uu,** i ve j kargasının arasındaki ucuş uzaklığını; **m,** j kargasının konumunu ifade etmektedir.   Capturdddsaassdae   * + Karga j, karga i’nin kendisini takip ettiğini bidiği varsayılıyor. Bu durumda karga j, karga i yi farklı bir konuma götürecektir. Karga i ‘nin yeni konumu rast gele seçilir.   + Kargaların takip edilip edilmediğini bulmak için **farkındalık olasılığı** kullanılır. * Hafıza Güncelleme   + Takip aşamasında konumları güncellenen kargaların, hesaplanan yeni uygunluk değerleri hafızalarındaki uygunluk değerleri ile karşılaştırılır. Yeni uygunluk değeri hafızadan daha iyi ise güncelleme yapılır. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 01.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 02.07.2019 |
| **SSA DSK Inceleme**   * **DAĞILIM**   + Popülasyonun oluşturulması düzgün dağılmış rastgele sayılar ile yapılmıştır [rand] * **SEÇİM**   + En iyi besin kaynağı her zaman hafızada tutulur. Lider ve izleyici salplerin konumları güncellendikten sonra uygunluk değerleri tekrardan hesaplanır. Hafızadaki uygunluk değerinden daha iyi olan sonuçlar hafızaya kaydedilerek liderin yönü değiştirilir. * **KONTROL**   + Algoritmada önemli olan parametreler c1,c2 ve c3 parametresidir. C2 ve C3 0-1 arasındadüzgün dağılmış rastgele sayılar ile yapılmıştır. En önemli olan C1 parametresi aşağıdaki formül ile hesaplanmaktadır. İterasyon başlangıçlarında yaklaşık değeri 2 olup sonlara doğru 0’a yaklaşmaktadır. Böylece global ve local arama arasındakı denge belirlenmiş olur.   Cap1112wqture  **CSA DSK Inceleme**   * **DAĞILIM**   + Popülasyonun oluşturulması düzgün dağılmış rastgele sayılar ile yapılmıştır [rand] * **Seçim**   + Yeni çözüm adaylarının seçimi için iki farklı yol bulunmaktadır.     - Verilen formül ile gerçekleştirilir. .İki nokta arasındaki fark çok büyük olduğu varsayıldığında, formülde uu ve r parametresinin küçük alınması ve üretilmesi, takip eden karganın konum güncellemeleri küçük olacağından komşuluk araması yapılacaktır. Büyük alınıp üretilmesi durumunda karganın konum güncellemeleri büyük olucaktır ve kısmi çeşitlilik sağlanacaktır fakat hedeflenen karga aynı olduğundan belli iterasyon sonuçunda yerel min tuzaklarına takılma olasılığı bulunmaktadır.     - Takip edildiğinin anlaşılması durumunda karganın yeni konumu düzgün dağılmış rastgele sayılar ile yapılmıştır [rand] * **Kontrol**   + CSA algoritmasında arama uzayını etkileyen iki parametre bulunmaktadır. Bunlar bir karganın takip edildiğini anlaması için kullanılan **farkındalık olasılığı** ve iki karga arasındaki **uçuş uzaklığıdır.** Bu parametreler algoritma başlangıcında belirlenip, iterasyon boyunca sabit kalmaktadır. İncelenilen algoritmada bu parametrelerin değerleri sırayla 0.1 ve 2 alınmıştır. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon MSA İnceleme | TARİH : 03.07.2019 |
| Güve sürü algoritmasında, optimizasyon probleminin çözümü ışık kaynağının konumu olarak belirlenmiştir ve çözümün amaç fonksiyonu da ışık kaynağının parlaklık oranı olarak tanımlanmıştır. Algoritma içerisinde tanımlanan güve sürüsü **üç grup halinde değerlendirilmiştir**. Bunlar:   * **Yolbulucular =>** sürü içerisinde küçük bir grubu (Np) oluştururlar ve optimizasyon evreni içerisinde yeni alanlar keşfedeler. Bu grubun asıl amacı problem içerisindeki en iyi sonucları yani en iyi ışık kaynaklarının yerlerini belirleyip asıl sürüyü buraya yönlendirmektir. * **Arayıcılar =>** yolbulucular tarafından belirlenen ışık kaynakları etrafında rastgele sarmal şekilde hareket eden sürü içerisindeki güve grubudur * **İzleyiciler =>** arayıcılar tarafından belirlenen optimizasyon evrenindeki en iyi sonuca doğru hareket eden güve grubudur.   Sürü içerisindeki en iyi uygunluk değerleri, yolbulucuların pozisyonu olarak tanımlanır ve bu tanımlama bir sonraki güncellenmiş döngü için kılavuz görevi görür. Daha sonra, ikinci ve üçüncü uygunluk değerleri ise sırasıyla arayıcılar ve izleyiciler olarak adlandırılır. Güve sürü algoritması dört aşamalı olarak çalıştırılmıştır;   * **Başlangıç Aşaması**   + Ajanlar rastgele biçimde üretilir.   + Sürüdeki her bir güvenin türü, hesaplanan uygunluk değerlerine göre belirlenir. * **Keşif araması**   + Güveler, iyi gibi görünen alanlarda yoğunlaşabilir ve bu durum durgunluğa sebep olur. Erken gelişen yakınsamaya engel olmak ve çeşitliliği geliştirmek için sürüde bulunan bir grup güve, daha az kalabalık olan alanları keşfetmek için seçilirler. Bu seçilmiş grup yolbulucular olarak tanımlanır ve birbirleriyle iletişime geçerek (geçiş işlemi) konumlarını günceller. Konum güncelleme işlemi, lévy mutasyon olarak adlandırılan güvelerin uzun mesafe uçma yetenekleri ile ilişkilendirilir ve **beş başlık altında tanımlanabilir**.     - **Ca213ptureCapt223ureÖnerilen geçiş noktaları için çeşitlilik indeksi.=>** np yolbulucu güvelerin sayısı olmak üzere verilen formüller ile ơ ifadesi hesaplanır. Daha sonra bağıl yayılma ölçüsü olan değişim kaysayısı µ, ơ ortalaması alınarak bulunur. Düşük yayılma derecesine sahip yolbulucu güvelerin herhangi bir bileşeni geçiş noktası grubuna (Cp) Kabul edilir ve uyulması gereken gereksinim “ơ <= µ” ile tanımlanır.     - **Levy uçuşu =>** α-kararlı dağılımına dayanan ve farklı ölçülerde adım büyüklüğü kullanarak büyük boyuttaki mesafelerde hareket etme kabiliyeti olan rastgele işlemlerdir. Gerekli formüller aşağıda verilmiştir.   Captur2eCapture  Capt23ure | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon MSA İnceleme | TARİH : 04.07.2019 |
| * + - Caddddssdsapture **Levy mutasyon fark vektörleri =>** Host vektörün seçilen elemanlarını bozarak bir alt-iz vektörü oluşturur. Bu tarz bir alt-iz vektörü oluşturmak için mutasyon yöntemi aşağıdaki şekilde kullanılabilir;   **Lp ifadeleri mutasyon ölçekleme katsayısı olarak kullanılan ve Levy yardımı ile üretilmiş değişkenlerdir.**   * + - C:\Users\xmkati\AppData\Local\Microsoft\Windows\INetCache\Content.Word\izDD.PNG**Sürü çeşitliliğine dayanan uyarlanabilir geçiş işlemi =>** Tamamlanan iz çözümlerini başarmak yerine, her yolbulucu çözümü (host vektörü) geçiş işlemi kanalıyla, alt-iz vektörünün birleşik mutasyona uğramış değişkenleri kullanılarak konumunu günceller. Tamamlanan iz çözümü yanda verilen şekildeki gibi tanımlanabilir.     - **Seçme yöntemi =>** Bir önceki aşama bittikten sonra, tamamlanan iz çözümünün uygunluk değeri hesaplanır ve ilgili host çözümü ile karşılaştırılır. En uygun çözümler bir sonraki iterasyon için seçilir.   + Konum güncelleme işlemi tamamlandıktan sonra yolbulucu grubunun olasılık değeri Pp aşağıdaki formüller ile bulunur.   Captufdöjgdre  33sdCapture | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon MSA İnceleme | TARİH : 05.07.2019 |
| * **Dikey Yönelim**   + MSA’da yolbulucu grubundan sonra en iyi parlaklık yoğunluğuna sahip güveler arayıcılar (Nf) olarak isimlendirilmişti. İterasyon boyunca bu grubun üye sayısı aşağıdaki formül ile azaltılıp izleyici grubunun üye sayısı arttırılarak algoritmanın sonlarına doğru sömürü özelliğinin artması sağlanmıştır.   ffCapturffe   * + 3443CaptureArayıcı grubu belirtilen spiral şeklindeki uçuş yörüngesine göre konumlarını günceller. Bu güncelleme işlemi matematiksel olarak aşağıdaki gibi ifade edilebilir. * **Göksel Navigasyon**   + İzleyici güveler iki gruba ayrılarak konum güncellemeleri farklı yapılmaktadır.     - Caure**Grup1. Gauss yürüyüşü =>** Gauss olasılıksal dağılımı, rastgele örneklerin dağılımını sınırlama kabiliyetine sahiptir. Bu yürüyüş aşağıdaki formül ile gerçekleştirilir.     - **Grup2. Anlık hafıza yöntemi =>** Bu tarzda bir hafıza, bir nesildeki bilgileri bir sonraki nesle aktarmak için birçok optimisazyon algoritmalarında bulunur. PSO algoritmasının hız formülüne benzeyen ifade aşadağıda verilmiştir.     Cssssapture | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 08.07.2019 |
| Close Air Support Request raporunun projeye ekleme görevi alındı. Ekleme işlemine başlamadan önce projede yer alan Observation Report, Overlay Report ve Position Report classları ve VMF açıklamaları incelendi. İncelenen VMF açıklamalarına örnek görsel aşağıda verilmiştir.  Verilen görsel 1 sayfa olacak şekilde, Observation Report için 3 sayfa, Position Report için 2 sayfa ve Overlay Report için 14 sayfa gözden geçirilip çalışma günü tamamlanmıştır. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 09.07.2019 |
| Close Air Support Request raporunun yazılması için VMF açıklamaları (6 sayfa) incelemenye başlandı. Örnek açıklama sayfası aşağıdaki gibidir.  Açıklamalar class’a dönüştürülürken bazı kurallar geçerlidir. Bu kurallar;   * Base class oluşturulup Gri referanslarının başladığı yerlerde base classın içersine child classlar oluşturulur. Bu child classların nesneleri base class’da liste şeklinde tutulur. * Gri referanslarının başlangıç ve bitiş noktaları Repeat Code alanında R ifadelerinin başlangıç ve bitişleridir. * Cat M ve X ifadeleri oluşturulan base class’ta olma zorunluluğu bulunmaktaadır. M alanları doldurulması zorunlu fieldler, X ler ise null gönderilebilecek fieldleri temsil etmektedir. * FPI, GPI, GRI ve FRI referansı haricindekiler referans adları aynı olacak şekilde class’a eklenir.   Verilen kurallara ve önceden yazılmış rapor sınıfları tekrardan incelenerek CloseAirSupportRequest sınıfı oluşturulmaya başlandı. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | DOOB Tactical Projesi | TARİH : 10.07.2019 |
| CloseAirSupportRequest classın’da eksik kalan kısımlar tamamlandı. Incelenen 6 sayfalık dökümantasyonun veri tabanına aktarılması gerektiğinden, ilgi rapor tablosunun sutunları incelendi. Tüm alanları elle girmek yavaş ve karmaşık olduğundan aşağıdaki örnek sql sorgusu yazılarak ekleme işlemlerine devam edildi.    Veri tabanına ekleme işlemi yapılırken CloseAirSupportRequest classın’a eklenmeyen fieldler eklenip, adlandırmada yapılan yanlışlıklar düzeltildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 11.06.2019 |
| CloseAirSupportRequest classında ve veri tabanına ekleme/düzeltme işlemleri bitirildi. Yeni görev oluşturulan class sisteme entegre edilmesidir. Gizlilik söz konusu olduğundan işlem adımlarının bazıları aşağıdaki gibidir;   * CloseAirServis interfacesinin yazılması + * CloseAirServisImp sınıfının yazılması + * CloseAirEntity sınıfının yazılması + * Arayüze CloseAirSupportRequest raporunun açılabilmesi ve izlenmesi için ilgili bölümlerin eklenmesi + * Haritaya eklenecek şeklin tipinin veri tabanından değiştirilmesi / eklenmesi + * Eklenen yeni tipdeki şeklin harita üzerinde çizdirilmesi için ilgili kodlara eklemeler ve değişikliklerin yapılması.+ * Şekil çizildiğinde oluşan entity’nin tipine göre nesnenin oluşturulması.+ * Çizilen şekil kayit edildiğinde, ait olduğu rapor servisinide tetiklediği template fonksiyonların yazılması+ * Rapor ile ilgli veri tabanında kayitların veya güncellemelerin yapılması+ * İlgili enum tiplerine eklemelerin yapılması+ * Eşleşmiş cihazlara oluşturulan raporun gönderilmesi+ * Gelen bilginin hangi rapora ait olduğunun tespit edilmesi * Rapor tipi belli olan bilginin işlenip veri tabanına kayıt edilmesi     Gerçekleştirilen işlemler önceden yazılmış olan MEDEVAC raporundan esinlenerek başladı. İlk olarak CloseAirServis interfacesi yazıldı. CloseAirServisImp sınıfına implementasyonu gerçekleştirildi. Sınıfın adları Save ve Create olan temel iki fonksiyonu bulunmaktadır. Save metodunda CloseAirEntity nesnesinden gelen bilgiler ile CloseAirSupportRequest nesnesi oluşturulur. CloseAirEntity nesnesi harita üzerinde çizilen şekli temsil etmektedir. Set edilecek fieldler tespit edilip, set edilmesi için gereken fonksiyonlar oluşturuldu. CloseAirEntity sınıfı oluşturulup eklenecek şeklin tipi için veri tabanında gerekli değişiklikler yapılıp çalışma günü bitirildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 12.06.2019 |
| Yapılan işlemler maddeler halinde aşağıdaki gibidir.   * Arayüze ilgili bölümler eklendi * Şekil adları veri tabanında tutulmaktadır. Uygulama açıldığında, çizilen şekillerin hangi rapora ait oldukları ve adları veri tabanından çekilir. Veri tabanına yeni eklenen raporun şekli dün yapılan çalışmalarda eklenmişti. Çizim sırasında çekilen şekil verilerisinin doğruluğunu kontrol eden kod bloklarında düzenlemeler ve eklemeler yapıldı * Çizimi yapılan şeklin rapor tipi alınarak, o tipe özgü entity oluşturulur. CloseAir için gerekli eklemeler yapıldı * Çizilen şekil-entity kayit butonu ile kayit edildiğinde çağrılan template fonksiyonları yazıldı. * İlgili enum tiplerinde güncellemeler yapıldı * Oluşturulan rapor entitysinin veri tabanına eklenmesi için gereken kodlar yazıldı. History tabloları oluşturuldu | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 16.07.2019 |
| Yapılan işlemler maddeler halinde aşağıdaki gibidir.   * Eşleşmiş cihazlara CloseAir raporunun açık olması durumunda gönderilme işlemi tamamlandı. UDP modulü ile bağlantılar tamamlandı. * Gelen bilginin hangi rapora ait olduğu bilgisi kontrol edecek ve gerekli yönlendirmeleri yapacak fonksiyonlar yazıldı * Tipi belli olan raporun işlenip ilgili taplolarla güncelleme veya ekleme yapılması için gerekli bağlantılar sağlandı. * Yapılan işlemlerin ardından uygulanın ilk testi gerçekleştirildi. CloseAir raporunun haritaya çizilmesi için gereken butonların birisinin click özelliği disable olduğu belirlenip nedeni araştırıldı. Bazı enum tiplerinde eklemelerin gözden kaçırıldığı fark edilerek bu hata düzeltilip çalışma günü bitirildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 17.07.2019 |
| CloseAirServisImp sınıfında bulunan save fonksiyonu yazılmaya başlandı. Bu fonksiyon CloseAir raporu alındığında tetiklenmektedir. Gelen entitynin veri tabanında daha önceden varsa güncelleme yoksa kayıt işlemlerini gerçekleştirecek kodlar eklendi. Arayüze CloseAir raporunun adının basılması içi eklenen labelin setlenmesinde hata bulundu. Hatanın nedeninin labelin idsinde yapılan hatadan kaynaklandığı bulundu. Gerçekleştirilecek testler için veritabanının backubı ve gerekli jar dosyaları alınıp tablerler hazır hale getirilip çalışma günü bitirildi. | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |

|  |  |  |
| --- | --- | --- |
| KISIM |  | YAPRAK NO : |
| YAPILAN İŞ | Meta Sezgisel Optimisazyon Araştırma | TARİH : 28.06.2019 |
| SSA algoritmasının ana ilham kaynağı salp canlılarının okyanustaki hareket ve besin arama yöntemleridir. Salp canlıları derin okyanuslarda genellikle salp zinciri denilen bir sürü özelliği gösterir. Zincirin başında salp lideri yer alır ve zinciri yönlendirir.  SSA matematiksel modeli için oluşturulacak popülasyon iki gruba ayrılmıştır: lider ve takipçiler.    CSA algoritmasının çalışma adımları ise;   * Başlangıç Aşaması   + Salp sürüsü rastgele biçimde üretilir   + Üretilen salp sürüsü uygunluk değerine göre sıralanır. Minimizasyon problemleri için en küçük uygunluk değerine sahip salpler sıralama sonucu en başa gelicek ve lider olucaktır. * Zincirin yönlendirilmesi – Konum güncelleme   + Konum güncelleme iki farklı şekilde yapılmaktadır     - **Liderin pozisyonu** aşağıdaki denklem ile güncellenmektedir. Bu güncelleme besin kaynağına (Bj) göre yapılmaktadır.   Cadddasdapture   * + - **Takipçilerin posizyonu** aşağıda verilen denklem ile güncellenir.   Cassdapdature | | |
| KONTROL SONUCU | | |