

Nokta Etiketi Yerleştirme Problemi için Hızlı Sezgisel Eniyileme Yöntemleri Fast Heuristic Optimization Methods for Point Label Placement Problem

Orkun Akile¹, Erdoğan Sevilgen¹

¹Bilgisayar Mühendisliği Bölümü, Gebze Teknik Üniversitesi
{oakile, sevilgen}@gtu.edu.tr

Özetçe—Nokta etiketi yerleştirme problemi harita tabanlı sistemler gibi grafiksel görüntüleme sistemlerinde noktasal öğelere ait etiketleri çakışmalarını asgariye indirecek şekilde yerleştirme ve böylece okunurluğu azamiye çıkarma problemidir. Genel olarak bu problemin NP-Zor algoritma karmaşıklık sınıfında olduğu ispatlanmıştır. Probleme çözüm üretmek, dinamik ve gerçek zamanlı sistemler için böyle sistemlerin yaygın ve geniş çaplı kullanıma potansiyelinden dolayı giderek önem kazanmaktadır. Bu çalışmada hem görüntüleme tercihlerinin değişebildiği, hem de noktasal öğelerin hareket edebildiği dinamik ortamlar için hızlı ve iyi kalitede etiket yerleştirmeyi sağlayabilecek sezgisel eniyileme yöntemleri önerilmektedir. Önerilen yöntemler iki aşamalı olup, hırslı inşa ve yerel aramadan oluşmaktadır. Yöntemlerin performansları çözüm kalitesi ve çalışma zamanı açısından incelenmiştir. Önerilen elemeli, iki kriterli yöntem hem bu çalışmadaki diğer yöntemlerden, hem de mevcut benzer çalışmadakinden daha iyi performans göstermektedir.

Anahtar Kelimeler—*kombinatorial eniyileme, sezgisel algoritmalar, nokta etiketleme*

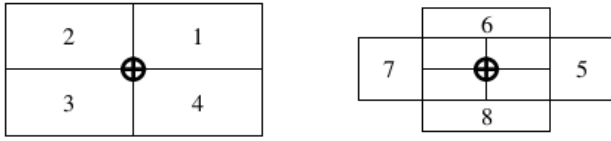
Abstract—Point label placement problem is the problem of labeling points in graphical display systems such as map-based ones in order to minimize conflicts of labels and thus to maximize legibility. Algorithmic complexity of the general problem is proven to be NP-hard. Addressing this problem is becoming more important in dynamic and real-time environments, where both display properties or point elements are subject to change, since those systems have potential use widely and in large scale. In this study, heuristic optimization methods are proposed to facilitate fast and good-quality labeling in such dynamic environments. The methods are two-phased and comprised of several combinations of greedy construction and local search methods. Performances of the methods are examined in terms of solution quality and run time. The eliminative, two-criteria method outperform other methods in this study and the methods proposed in the literature.

Keywords—*combinatorial optimization, heuristic algorithms, point labeling*

I. GİRİŞ

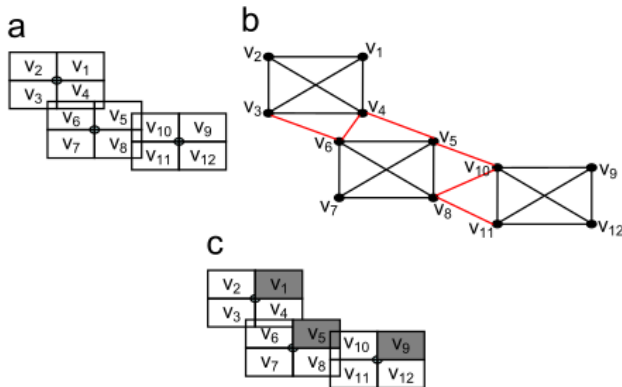
Nokta etiketi yerleştirme problemi genel olarak noktasal öğelere ait etiketleri aday konumlardan birine diğer etiketlerle veya noktasal öğelerle çakışmayı asgariye indirecek, dolayısıyla okunurluğu azamiye çıkaracak şekilde yerleştirme problemidir. Başlangıcından itibaren çoğunlukla “noktasal bileşen etiketleme problemi” adıyla haritacılık alanı ile ilgili olarak üzerinde yıllardır çalışılmaktadır. Çünkü etiket yerleştirme temel ve insanlar tarafından yapılması oldukça zahmetli bir iştir. Haritalarda açıklık ve uyumu sağlamak için haritacılık alanda belirli etiketleme tercihleri tanımlanmıştır. Kartografik tercihler aday konum modeli ve konum değerlendirme ölçütleri belirlenmiştir [1]. Günümüzde haritaların dinamik olduğu veya hareketli öğelerin yer aldığı grafiksel sergileme sistemleri mevcuttur. Bu çalışmada dinamik sistemler amaçlandığından kartografik tercihler dikkate alınmamıştır.

Problemin aday konum modeli ve hedef fonksiyonu açısından farklı çeşitleri mevcuttur. İlk olarak aday konum modeline göre problem kesikli veya sürekli (kayan etiketler) olabilmektedir. İkincisi, konum boyutları açısından tüm noktalar için etiket boyutu aynı veya farklı olan problemler mevcuttur. Bir diğeri, hedef fonksiyonunun bazı noktaları etiketlememe pahasına azami çakışmasız etiket yerleşimi yapması veya tüm noktaları etiketleyerek çakışma sayısını asgariye indirmesidir. Ayrıca çakışmalar çakışma alanı büyüklüğünü veya sadece varlığı/yokluğu ölçüsünden hesaplanabilir. Christensen v.d. [2] tarafından ortaya konulan 4 ve 8 konumlu kesikli konum modeli aşağıda verilmiştir. Burada kartografik tercihlere de etiketler üzerindeki numaralarla yer verilmiştir. Bu çalışmada 4 konumlu, merkezci model esas alınmıştır. Hedef fonksiyonu ise tüm noktaları etiketlenmesi sonrasında çakışmasız etiket sayısını en fazla yapmaktır. Çakışma alanları hesaba katılmamıştır. Literatürdeki çalışmalar çoğunlukla bu çalışmada ele alınan problem tipi üzerinedir.



Şekil 1: Kesikli ve Merkezci Aday Konum Modeli

Genel etiket yerleştirme problemi ve bu problemin yaygın olarak kabul edilen tipleri NP-Zor algoritmik karmaşıklık sınıfındadır [3]. Bu yüzden bu probleme verimli bir şekilde çözüm üretmek için iyi yaklaşık yöntemlere ihtiyaç duyulmaktadır. Ayrıca problem geometrik yanından soyutlanarak, salt kombinatoriyal eniyileme problemine dönüştürülmüştür [2]. Geometriden soyutlanmış problem çakışma çizgesi ile temsil edilmektedir. Etiketlerin yerleştirilebileceği her bir konum çizgede bir düğüm ve konumlar arasında çakışma ise, ilgili iki düğüm arasındaki bir kenar ile gösterilmiştir. Örnek bir çakışma çizgesi aşağıda verilmiştir [11]:



Şekil 2: Örnek Çakışma Çizgesi a. Nokta ve Aday Konumlar b. Çakışma Çizgesi c. Optimal bir Çözüm

Bu problem için yaklaşık çözüm üreten yöntemler içerisinde sezgisel yöntemler ön plandadır. Sezgisel yöntemler temel ve meta-sezgisel yöntemler olarak ikiye ayrılmaktadır. Meta-sezgisel yöntemler temel sezgisel yöntemleri bir organizasyon dahilinde içeren üst-yöntemlerdir. Eniyileme yöntemleri ayrıca durağan veya dinamik yöntemler olarak da sınıflandırılabilir. Dinamik eniyileme yöntemleri değişen durumlara uyarlanma veya tahmin gibi kabiliyetlerle daha iyi performans sağlayabilirler.

Nokta etiketi yerleştirme konusundaki çalışmalar çoğunlukla durağan haritaları hedeflediğinden durağan eniyileme yöntemleri üretilmiş ve olabildiğince yüksek etiketleme kalitesine ulaşmaya çalışılmıştır. Bu yüzden bu

yöntemler dakikalar mertebesinde çalışma zamanı gerektirmektedir. Ancak durağan eniyileme yöntemleri yeterince hızlılarsa, belirli zaman aralıkları ile uygulanarak, dinamik problemin çözümü için uygulanabilir. Fakat nokta etiketi yerleştirme için yapılan çalışmaların çoğundaki dakikalara ulaşabilen zaman maliyetinin kabul edilebilir olmadığı açıktır. Bu sebepten çözüm kalitesinden biraz ödün veren fakat hızlı çalışan yöntemler tercih edilmelidir. Hızlı yöntemler konusunda az da olsa çalışmalar mevcuttur. Bu çalışmada da çeşitli durağan hızlı yöntemler önerilmektedir.

II. LİTERATÜR İNCELEMESİ

Nokta etiketi yerleştirme probleminin NP-Zor karmaşıklık sınıfında olduğu kanıtlanmıştır [3].

Problem için bu güne dek üretilen çözüm yöntemleri matematiksel modeller, yorucu arama (exhaustive search), yerel arama, hırslı inşa kategorilerinde yer almaktadır. Çalışmalardan bazıları çakışmasız azami etiket sayısı problemini, bazıları ise asgari çakışma sayısı problemini ele almıştır. Ayrıca az sayıda olmakla birlikte etiketleme çözüm kalitesinden ödün verip, hız kazanmayı amaçlayan yöntemler ortaya çıktığından, çalışmaları yüksek kaliteli yöntemler ve hızlı yöntemler olarak ikiye ayırabiliriz.

Yüksek kaliteli yöntemler kategorisinde, tam arama yöntemleri problemin üstel doğasına karşı sezgisel olarak ilerletilseler bile verimli çözümler üretmediğinden, bu tip çözümler artık geçerliliğini yitirmiştir. Matematiksel modeller tam sayı programlama ve çizge yöntemlerini kapsamaktadır. Zoraster [4], Mauri v.d. [5], Ribeiro v.d. [6][7] tam sayı programlama çözümleri sunmuşlardır. Güncel yakın çalışmalar üst sıralarda çözüm kalitesi başarıları elde etmişlerdir. Strijk v.d. [8] azami bağımsız kümeyi bulmaya yönelik çizge yöntemleri uygulamışlardır.

Problem Christensen v.d. [2] tarafından kombinatoriyal eniyileme alanında genel ve kapsamlı bir biçimde sezgisel yöntemlerle ele alınmıştır. Bu çalışma sezgisel yöntemlere zemin kazandırmıştır ve öncüsü olmuştur. Yöntemlerin başarılarını karşılaştırılabilir olarak ölçmek için bir ölçüt veri kümesi tanımlanmıştır. Bunun yanısıra hırslı yöntem ve ısısal sağlama benzetimi (simulated annealing) yöntemlerini önermiştir.

Verner v.d. [9] maskeleye ile iyi yerleştirmelerin korunmasını sağlayan genetik algoritma uygulamışlardır. Yamamoto v.d. [10] tarafından önerilen Tabu arama yöntemi belirli sayıda son etiketleme değişikliğinin yapıldığı değişken boydaki tabu listesini hariç tutan bir yerel arama yöntemidir. Cravo v.d. [11] tarafından uygulanan GRASP yöntemi hırslı bir inşa sürecinde, belirli bir sayıdaki aday konumlar içerisinde rastgele seçim yaparak rastsalılık eklemiştir. Öbekleme ile arama yöntemi Rabello v.d. [12] tarafından yapılan çalışma yüksek kaliteli yöntemler kategorisinde şu ana kadarki son çalışma olup, aynı zamanda en başarılısı durumundadır. Bir çözüm üretici ile üretilen çözümler Hamming yakınlığına göre öbeklere ayrılmakta, sonra en başarı vaat eden öbek üzerinde yerel arama uygulanmaktadır.

Hızlı yöntemler kategorisinde ilk olarak Wagner v.d. [13] tarafından yapılan çalışmada çizge özelliklerine dayanan yöntemden bahsedilebilir. Bu çalışma azami çakışmasız etiket yerleştirme problemini ele almıştır. Yöntemin ilk aşamasında üç çizge kuralı özyineli olarak uygulanmakta ve çakışmasızlık garantisi edilmektedir. İkinci aşaması ise ilk adımda etiketlenemeyen noktaları için etiket yerleşimini hedefleyen, her adımda etiketlenecek bir aday konumu eleyen yerel arama yöntemidir. Roy v.d. [14] tarafından yapılan çalışmada yine çizge tabanlı bir yöntem olarak "Plane Sweep" yöntemi uygulanmıştır.

Bu kategoride bizim çalışmamızla yakınlık açısından Yamamoto v.d. [15] tarafından yapılan çalışmadan özellikle bahsetmek gerekir. Bu çalışma çakışmasız etiket sayısını en fazla yapma problemini ele almıştır. Bu çalışmadaki yöntem hırslı inşa ve yerel arama olmak üzere iki aşamalıdır. Hırslı inşa aşamasında her adımda potansiyel çakışma sayısı en az olan bir konum seçilmektedir. Bu seçim yapıldıktan seçilen konumla çakışan diğer konumlar da aday konum olmaktan çıkarılmaktadır. Bu işlemin sonucunda etiketlenememiş noktalar ise potansiyel çakışma sayısı en az olan konumlar bu sefer onlarla çakışanlar çıkarılmadan seçilmektedir. Yöntemin ikinci aşaması tek değişimli yerel aramadır. Bu çalışmada yüksek kaliteli yöntemlerden uzak olmayan ve onlara göre oldukça hızlı sonuçlar elde edilmiştir. Performans çözüm kalitesi ve zaman maliyeti olarak verilmiştir.

III. YÖNTEMLER

Bu çalışmada önerilen yöntemler hırslı inşa ve yerel arama olmak üzere iki aşamadan oluşmaktadır. Hırslı algoritma bir çözümü sıfırdan oluştururken her adımda yerel olarak en iyi seçimi yapan ve bir problem çözme sezgiselini izleyen bir yöntemdir. Yerel arama algoritmaları komşuluk uzayında bir çözümünden bir diğerine yerel bir en iyi çözüm bulunana veya belirli bir çalışma miktarı kısıtına ulaşıncaya kadar değişiklikler yaparak geçer. Bu yöntemler temel sezgisel yöntemlerdir. Yöntemlerin çeşitliliği hırslı aşamadaki değerlendirme çeşitliliğinden kaynaklanmaktadır. Her bir hırslı yöntem varyantı ile aynı yerel arama yöntemi birlikte çalışmaktadır. Hırslı yöntemler problemin doğasındaki temel bazı özellikler ayrı ayrı veya birlikte kullanılarak elde edilmişlerdir.

Burada potansiyel çakışma sayısı kavramını tanımlamak gerekir. Bir aday konumun potansiyel çakışma sayısı diğer aday konumlarla veya seçilmiş konumlarla olan çakışma sayıları ile hesaplanan ve bir niceliktir. Aday konumun seçilmesinde doğrudan etkilidir. Nasıl hesaplandığı bu bölümde ileride anlatılacaktır.

Hırslı aşamada iki ana yaklaşım vardır. Bunlardan birincisi seçimli, diğeri elemeli yaklaşımdır. Seçimli yaklaşımda her bir adımda bir aday etiket konumu etiketlenmek üzere seçilmekte, o noktaya ait diğer konumlar çıkarılmaktadır. Elemeli yaklaşımda her adımda bir aday etiket konumu elenmektedir. Noktaların yalnızca birer aday konumu kalana kadar işlem adımları devam etmektedir. p

adet noktanın her biri için olası konum sayısı n ise, seçimli yöntemdeki işlem adımı sayısı p, elemeli yöntemde işlem adımı sayısı $(n-1) \times p$ olacaktır.

Yaklaşımların her biri için ise, 3 strateji mevcuttur. Seçmeli yaklaşım için ilk strateji potansiyel çakışma sayısı en az olan noktayı etiketlemektir. Yani seçilen konumun potansiyel çakışma sayısını asgariye indirmektir. İkincisi potansiyel çakışma sayısını en fazla azaltacak noktayı seçip, etiketlemektir. Yani seçilen noktanın en küçük potansiyel çakışma sayısına sahip konumu dışındaki konumlarının potansiyel çakışma sayılarını azamiye çıkarmaktır. Üçüncü yaklaşım ise ilk iki stratejiyi birlikte ağırlıklandırıp çoklu kriter olarak uygulamaktır.

Elemeli yaklaşım için de önerilen stratejiler benzer şekilde elenenlerin potansiyel çakışma sayısını azamiye çıkarmak, seçilebilecek olanların potansiyel çakışma sayılarını asgariye indirmek fikrini ve bunların ayrı bir yöntemde çoklu kriter olarak kullanılmasını esas almaktadır.

Detaylı ifade etmek gerekirse; ilk strateji elenecek bir konumun potansiyel çakışma sayısını azgariye indirmektir. İkinci strateji eleme yapılacak noktada kalacak olan konumların potansiyel çakışma sayıları ortalamasını asgariye indirmektir. Üçüncü strateji, yine ilk iki stratejiyi birlikte ağırlıklandırıp, çoklu kriter olarak değerlendirmektir.

Bu çalışmada potansiyel çakışma sayısı değerini hesaplamak için yeni bir yol önerilmektedir. Bu yolda aday konumların seçilme olasılıkları kadar potansiyel çakışma yaratacakları gerçeği esas alınmıştır. Aday konumların seçilme olasılıkları ait oldukları noktanın kalan aday konum sayısı ile ters orantılıdır. Bu değer çakışma ağırlığı olarak adlandırılmıştır. Her bir aday konumun potansiyel çakışma sayısı mevcut durumda çakıştığı aday konumların çakışma ağırlıkları toplamıdır. Eğer bir aday konum seçilmiş ise değeri 1 olacaktır.

Bu durumda seçimli yaklaşımda, bir noktaya için seçim yapıldığında diğer aday noktalar çakışma çizgesinden çıkarıldığı için aday konumların çakışma ağırlıkları 1 veya $1/n$ olacaktır. Elemeli yaklaşımda her bir adımda 1 aday konum çıkarıldığından aday konumların çakışma ağırlıkları 1, $1/2$, ..., $1/n$ değerlerini alabilir.

Seçimli yaklaşım için genel algoritma aşağıda verilmiştir:

1- Etiketlenmemiş nokta kalmayana kadar tekrarla:

1.1- Stratejiye göre seçim yapılacak noktayı bul.

1.2-Noktanın potansiyel çakışma sayısı en düşük olan konumunu etiketlemek için seç. Yani diğer $n-1$ konumu çakışma çizgesinden çıkar.

1.3-Noktayı etiketlenecek noktalar listesinden çıkar.

Seçmeli yaklaşımdaki stratejilerin seçim yapılacak noktayı bulma adımları aşağıdadır:

Strateji 1 – Seçileni asgariye indirme: Noktalardan aday konumlarının en küçük potansiyel çakışma sayısına göre en küçüğünü al.

Strateji 2 – Seçilmeyenleri azamiye çıkarma: Noktalardan aday konumlarının en büyük n-1'inin toplam potansiyel çakışma sayısına göre en büyüğünü al.

Strateji 3 – 2 Kriterli: Noktalardan aday konumlarının en küçük potansiyel çakışma sayısından, diğer konumların toplamının ağırlıklı farkına göre sıralamada en küçüğünü al.

Elemeli yaklaşım için genel algoritma aşağıda verilmiştir:

1- Etiketlenmemiş nokta kalmayana kadar tekrarla:

1.1-Stratejiye göre eleme yapılacak noktayı bul.

1.2-Noktanın potansiyel çakışma sayısı en yüksek olan konumunu ele. Yani çakışma çizgesinden çıkar.

1.3-Eğer noktaya ait yalnızca 1 adet aday konum kalmışsa, noktayı etiketlenecek noktalar listesinden çıkar.

Elemeli yaklaşımdaki stratejilerin seçim yapılacak noktayı bulma adımları aşağıdadır:

Strateji 1 – Eleneni azamiye çıkarma: Noktalardan aday konumlarının en büyük potansiyel çakışma sayısına göre en büyüğünü al.

Strateji 2 – Kalanları asgariye indirme: Noktalardan aday konumlarının en küçüğü dışındakilerin potansiyel çakışma sayısı ortalamalarına göre en küçüğünü al.

Strateji 3 – 2 kriterli: Noktalardan aday konumlarının en büyük potansiyel çakışma sayısından, diğerlerinin ortalamalarının farkına göre sıralamada en büyüğünü al.

Yöntemlerin 2. aşaması yerel aramadır. Yerel aramada bir durumun komşuluğunu tanımlamak gerekir. Burada durum mevcut bir çözümdür. Yöntemimizdeki komşuluk mevcut bir çözümde 1 etiket yerleşimi değişikliği yapılarak elde edilebilecek çözümlerdir. Dolayısıyla bir çözümün komşuluğu $p \times (n-1)$ sayıda seçenekten oluşmaktadır. Ayrıca yerel aramada komşuluk uzayı içerisinde nasıl gezileceği, nasıl seçim yapılacağı tanımlanmalıdır. Yöntemimizde komşuluk uzayı sırayla taranmakta, mevcut çözümden daha iyi olan ilk yeni çözüm seçilmektedir. Yerel arama daha iyi bir çözüm elde edilemeye kadar sürdürülmektedir.

Yerel arama yöntemimizde tamlık özelliği vardır. Yani bir çözüm elde edildiğinde o çözüme ait tüm komşuluk uzayında arama yapılabilir. Fakat burada yerel arama ile ilgili bir zaman maliyeti iyileştirmesi yapılmıştır. Yeni bir çözüm elde edildiğinde önceki çözüme ait komşuluk uzayından kalan kısma, yeni çözüme geçişle birlikte meydana gelen değişiklikler eklenerek, değişiklik meydana

gelmeyen durumların yeniden aramaya dahil edilmesinin önüne geçilmiştir.

Yerel arama algoritması aşağıdaki gibidir:

1-Girdi alınan çözümü mevcut çözüm olarak ata.

2- Tüm noktaları taranacak noktalar kümesine ekle.

3-Taranacaklar kümesinde nokta kalmayana kadar aşağıdaki adımları tekrarla.

3.1-Kümeden bir nokta al ve kümeden çıkar.

3.2-Noktanın seçilmemiş her bir aday konumu için aşağıdaki adımları tekrarla

3.2.1- Eğer aday konum seçildiğinde çözüm iyileşiyorsa

3.2.1.1-Mevcut çözüme yeni çözümü ata.

3.2.1.2-Eski ve yeni seçilmiş konumla çakışma olan konumların ait oldukları noktaları taranacak noktalar kümesine ekle.

IV. DENEY

Yöntemler C++ programlama dilinde STL ve Boost kütüphaneleri kullanarak gerçekleştirilmiştir. Hırslı algoritmalarda her adımdaki azami veya asgari değerleri almak için fibonacci öncelik kuyruğu kullanılmıştır.

Noktasal öge etiketleme probleminde yöntemlerin performansını ölçmek ve karşılaştırabilmek için literatürde bir sınama girdi veri kümesi mevcuttur. Bu kümede 100, 250, 500, 750 ve 1000 adet noktalı girdilerin her birinden 25'er adet bulunmaktadır. Bu girdi kümesi belirli ve değişmez bir harita boyu varsayılarak, belirtilen sayıdaki noktaların bu alana rastgele yerleştirilmesiyle elde edilmiştir ve bir web kaynağında [16] sunulmaktadır. Yani bir girdideki nokta yoğunluğu, dolayısıyla çakışma sayısı nokta sayısı arttıkça artmaktadır.

Önerdiğimiz 6 yöntem bu girdi kümesi üzerinde çalıştırılarak performansı çözüm kalitesi ve çalışma zamanı cinsinden ölçülmüştür. Program Intel Core i7 4 çekirdekli 2,4 GHz saat frekansına sahip bir işlemci, 8 GB RAM ve 64 bit Linux işletim sistemi ortamında çalıştırılmıştır. Yöntemlerin ilk safhasının ve ikinci safhasının sonundaki çözüm kalitesi ve çalışma zamanı performansları her bir girdi için elde edilen değerlerin problem büyüklüğü bazında ortalamaları alınarak Tablo 1, Tablo 2, Tablo 3 ve Tablo 4'te verilmiştir.

Tablo başlıklarındaki S kısaltması seçmeli yaklaşımı, E kısaltması elemeli yaklaşımı belirtmektedir. Strateji numaraları yukarıdaki anlatıldığı sırayla kısaltmaların yanına yazılmıştır. Çözüm kalitesi performanslarında çakışmasız etiket oranı, zaman performanslarında harcanan zaman değerleri yer almaktadır.

Çözüm kalitesi açısından hırslı aşamada yaklaşımlar içerisinde 2 kriterli yöntemlerin diğer yöntemlerden daha başarılı olduğu görülmektedir. Ayrıca 2. yaklaşımdakilerin

ortalama olarak 1. yaklaşımlardan daha iyi olduğu tespiti yapılabilir. Bunun sebebi 2. yaklaşımdaki iterasyon sayısının daha fazla olmasıyla açıklanabilir. Dolayısıyla hızlı aşama yöntemlerinin en iyisi elemeli-2 kriterli (E3) yöntemdir.

Yerel arama aşaması sonuçları çözüm kalitesi bakımından incelendiğinde, bir önceki aşama olan hızlı aşamadaki sıralamanın büyük oranda korunduğu anlaşılmaktadır. Ayrıca yerel arama aşaması daha başarısız yöntemlerde daha büyük artış meydana getirmiştir. Bu durumlarda yerel arama aşamasının daha uzun süreler çalışması beklenmektedir. Yerel arama aşamasında da 2 kriterli yöntemler yaklaşımlar dahilinde diğerlerine göre daha iyi performans sergilemiş olup, yine elemeli-2 kriterli yöntem en iyi başarı sağlamıştır.

Zaman bakımından yöntemlerin performansı incelendiğinde, hızlı aşama yöntemlerinin zaman tüketimlerinin saniyenin epey altında olduğu ve S1 hariç diğerlerinininkilerin ise birbirine yakın olduğu gözlenmiştir. Nihai zaman tüketimleri olan yerel arama sonrası sonuçlara bakıldığında, küçük girdiler için nihai süreler arasında anlamlı bir fark bulunmadığı, büyük girdiler için (750 ve 1000 noktalık) elemeli yöntemlerin çoğunlukla seçmeli yöntemlerden kısa sürdüğü tespit edilmiştir. Bunun sebebi elemeli yöntemlerin hızlı aşamada daha yüksek performans göstermeleri ve nihayetinde yerel arama ile yerel asgari konuma daha çabuk ulaşmalarıdır. Hızlı aşamada elemeli yöntemlerin çok az daha fazla süre tüketmeleri, yerel arama aşamasından sonra fazlasıyla tazmin edilmiştir.

Yöntemlerin tamamlanma süreleri 1000 noktalı problem girdileri için 2 saniye civarında veya altındadır. 750 noktalı girdiler için ise 1 saniyenin altındadır. Gerçek zamanlı ve dinamik sistemler için kullanılmaya uygundurlar.

Çözüm kalitesi açısından en iyi sonuçları veren elemeli-2 kriterli yöntemin zaman açısından ortalama olarak diğerlerinden oransal olarak daha maliyetli olsa da, miktar olarak pratikte önemli derecede kötü olmayan zaman performansı sergilediği anlaşılmaktadır. Hatta 1000 noktalık girdiler için anlamlı derece bir fark meydana getirerek, en kısa sürede tamamlanmıştır.

Bu çerçevede elemeli-2 kriterli yöntemin zaman açısından önemli derecede daha kötü olmadığından, çözüm kalitesi açısından en iyi olduğundan pratikte kullanmak için en iyi yöntem olduğu söylenebilir. Bu yöntemin performansı ayrıca Yamamoto v.d. [15] tarafından yapılan çalışmadaki çözüm kalitesi sonuçlarıyla kıyaslanmıştır. Çünkü karşılaştırılan çalışmada aynı problem ele alınmaktadır ve önerilen yöntem temel hızlı aşama ve sonrasında yerel arama uygulayan, bizim önerdiğimiz yöntemlerle aynı kategoridedir. Tablo 5'te E3 kodlu yöntemimizle FALP kıyaslanmıştır. Elemeli 2 kriterli yöntemle E2C-FLP (eliminative and 2-criteria fast label placement – elemeli ve 2 kriterli hızlı etiket yerleştirme) adı verilmiştir.

Yöntemimizin çözüm kalitesi performansının diğer çalışmadaki yöntemden her bir problem boyu daha iyi veya

eşit olduğu anlaşılmaktadır. Bu sebepten yöntemimiz FALP adlı yöntemden genel olarak daha iyi çözüm kalitesi performansı sergilemektedir.

PROBLEM BOYU	YÖNTEMLER					
	S1	S2	S3	E1	E2	E3
100	100.00	99.76	100.00	100.00	100.00	100.00
250	99.94	98.66	99.84	99.90	99.94	100.00
500	98.73	94.42	98.87	98.53	98.98	99.49
750	93.64	87.14	94.60	94.46	95.00	96.84
1000	83.19	76.95	86.50	85.72	86.19	90.10

Tablo 1: Hızlı Aşama Sonrası Çözüm Kalitesi Performansları (Yüzde)

PROBLEM BOYU	YÖNTEMLER					
	S1	S2	S3	E1	E2	E3
100	100.00	100.00	100.00	100.00	100.00	100.00
250	100.00	99.84	99.94	99.90	100.00	100.00
500	99.25	98.13	99.22	98.87	99.22	99.52
750	95.59	93.43	95.85	95.67	96.00	97.11
1000	88.60	85.66	89.22	88.52	88.78	91.22

Tablo 2: Yerel Arama Aşaması Sonrası Çözüm Kalitesi Performansları (Yüzde)

PROBLEM BOYU	YÖNTEMLER					
	S1	S2	S3	E1	E2	E3
100	0.08	1.04	1.00	1.76	1.96	2.36
250	3.16	7.16	7.12	8.68	8.56	12.04
500	11.68	32.92	30.48	30.72	30.00	44.48
750	28.60	81.64	74.12	69.80	67.28	99.00
1000	56.92	160.52	142.80	131.60	129.92	185.68

Tablo 3: Hızlı Aşama Sonrası Zaman Performansları (ms)

PROBLEM BOYU	YÖNTEMLER					
	S1	S2	S3	E1	E2	E3
100	11.08	10.48	10.44	10.72	10.80	11.08
250	71.40	73.48	75.80	72.08	70.92	73.52
500	315.08	362.04	335.60	308.60	304.92	322.04
750	862.88	994.80	917.24	890.88	819.44	836.80
1000	2032.44	2210.28	1908.48	1949.68	1900.04	1764.96

Tablo 4: Yerel Arama Aşaması Sonrası Zaman Performansları (ms)

YÖNTEM	PROBLEM BOYU				
	100	250	500	750	1000
E2C-FLP	100.00	100.00	99.52	97.11	91.22
FALP	100.00	100.00	99.50	96.70	90.70

Tablo 5: Benzer Çalışma ile Çözüm Kalitesi Performansı Kıyaslama

V. SONUÇ

Bu çalışmada tüm noktaları etiketleyerek, azami çakışmasız etiket sayısı problemi hızlı çözümler üretmek üzere ele alınmıştır. Bunun için temel hırslı algoritma ve yerel arama aşamalarından oluşan genel yöntem çerçevesinde problemin doğasına özgü alternatif özellikler kullanılarak çözüm yöntemleri türetilmiştir. 2 yaklaşım dahilinde 6 adet çözüm yöntemi önerilmiştir. Her yaklaşım dahilinde birbirinin alternatifi 2 strateji ve bu 2 alternatif stratejinin birlikte kullanıldığı 2 kriterli 1 strateji bulunmaktadır.

Hırslı aşamada noktalar için alternatif değerlendirme işlevleri, 2 kriterlilik, potansiyel çakışma sayılarını olasılıksal olarak ağırlıklandırmak ve yerel aramada komşuluk uzayını elde ederken yeni bir önceki komşuluğa yalnızca değişenleri ekleyerek tekrarlı tarama yapmamak özellikleri, yöntemlerimizin önerdiği yenilikler olarak sıralanabilir.

Yapılan deneyler sonucunda çözüm kalitesi açısından en iyi performansı elemeli-2 kriterli yöntem göstermiştir. Bu yöntem zaman performansı açısından da diğer yöntemlerden pratikte önemli ölçüde daha kötü değildir, hatta yüksek boydaki problem girdileri için daha iyi olabilmektedir.

Bu yöntem ile nokta sayısı büyüklüğü 1000 olan problemler için 2 saniyenin altında, 750 ve altındaki büyüklükteki problemler için ise 1 saniyenin altında zaman tüketmiştir. Ayrıca %90 ve üzeri çakışmasızlık performansı sağlamıştır.

Elemeli-2 kriterli yöntem ayrıca literatürdeki benzer kategoride, temel hırslı algoritma ve yerel arama aşamalarından oluşan bir çalışma ile kıyaslanmıştır. Yöntemimizin bu çalışmadan daha iyi sonuçlar elde ettiği görülmüştür.

Sonuç olarak elemeli-2 kriterli yöntem zaman maliyeti ve çözüm kalitesi performansı dolayısıyla, gerçek zamanlı ve dinamik sistemlere makul kaliteli çözümler sağlamak için önerilmektedir. Ayrıca diğer yöntemler de geçerli yöntemler olarak sunulmaktadır.

VI. REFERANSLAR

- [1] Imhof, E., Positioning names on maps, *The American Cartographer*, 2(2):128-144, 1975
- [2] Christensen, J., Marks, J., & Shieber, S., An empirical study of algorithms for point-feature label placement, *ACM Transactions on Graphics*, 14(3), 203–232, 1995
- [3] Marks, J. and S. Shieber, The computational complexity of cartographic label placement, Technical Report TR-05-91, Harvard University, 1991
- [4] Zoraster, S., The solution of large 0-1 integer programming problems encountered in automated cartography, *Operations Research*, 38(5):752-759, 1990
- [5] Mauri, G. R., Ribeiro, G. M., & Lorena, L. A. N., A new mathematical model and a Lagrangean decomposition for the point-feature cartographic label placement problem, *Computers and Operations Research*, 37, 2164–2172, 2010
- [6] Ribeiro, G. M., Constantino, M. F., & Lorena, L. A. N., Um estudo sobre desigualdades válidas para o problema de maximização de rótulos livres, *XLI Brazilian symposium on operational research*, Porto Seguro, Brazil, 2009
- [7] Ribeiro, G. M., & Lorena, L. A. N., Lagrangean relaxation with clusters for point-feature cartographic label placement problems, *Computers and Operations Research*, 35, 2129–2140, 2008
- [8] Strijk T, Verweij B, Aardal K. Algorithms for maximum independent set applied to map labelling, Department of Computer Science, Utrecht University, Technical Report UU-CS-2000-22, 2000.
- [9] Verner, O. V., Wainwright, R. L., & Schoenfeld, D. A., Placing text labels on maps and diagrams using genetic algorithms with masking, *Journal on Computing*, 9(3), 266–275, 1997
- [10] Yamamoto, M., Câmara, G., & Lorena, L. A. N., Tabu search heuristic for point-feature cartographic label placement *Geoinformatica*, 6(1), 77–90, 2002
- [11] Cravo, G. L., Ribeiro, G. M., & Lorena, L. A. N., A greedy randomized adaptive search procedure for the point-feature cartographic label placement. *Computers and GeoSciences*, 34(4), 373–386, 2008
- [12] Rabello, R. L., Mauri, G. R., & Ribeiro, M. R., A Clustering search metaheuristic for the point-feature cartographic label placement problem, *European Journal of Operational Research* 234: 802–808, 2014
- [13] Wagner F, Wolff A, Kapoor V, Strijk T., Three rules suffice for good label placement, *Algorithmica* 30: 334–346, 2001
- [14] Roy S, Bhattacharjee S, Das S, Nandy SC, A fast algorithm for point labeling problem, 17th Canadian Conference on Computational Geometry (CCCG); Carleton University: Ottawa, 155–158, 2005
- [15] Yamamoto M, Camara G, Lorena L., Fast point-feature label placement algorithm for real time screen maps, *GeoInfo 2005*, Campos do Jordão. INPE: Sao Jose dos Campos, 2005
- [16] Test örnekleri <http://www.lac.inpe.br/~lorena/instancias.html>