

COĞRAFİ KONUM VERİLERİNİN YOL AĞIYLA EŞLEŞTİRİLMESİNDE GRİD SİSTEMİ ALGORİTMASI

Mustafa Kemal Sürmeneli¹, Esmâ Dilek², Mehmet Mert³

ÖZET

Araç trafiğinin yoğun olduğu şehirlerde, bir noktadan başka bir noktaya ulaşmak için en uygun güzergâhların arandığı, trafik yoğunluğunun sürekli ve canlı olarak gözlem altında tutularak, anlık hareketlilik bilgilerinin veri tabanlarında saklandığı; bu verilerin akıllı ulaşım çözümleri geliştirmek için kullanıldığı akıllı şehircilik çağında yaşıyoruz.

Anlık trafik koşullarına göre en uygun güzergâhların hızlı ve doğru olarak üretilebilmesi için seyir halindeki araçlardan ve mobil uygulama kullanıcılarından toplanan konumsal veriler önemli trafik veri kaynakları olup; bu verilerin, gerçek zamanlı trafik bilgilendirme sistemlerinde kullanılabilmesi için kentsel yol ağıyla ilişkilendirilmesi gerekmektedir. Bu çalışmada, sahadan toplanan gerçek zamanlı konumsal verileri, literatürde yer alan yöntemlere göre daha hızlı bir şekilde kentsel yol ağı ile eşleştirerek, trafik bilgilendirme sistemlerinde kullanıma hazır hale getiren algoritma hakkında bilgi verilmektedir. Geliştirilen algoritma ile sahadan toplanan konumsal veriler, yol ağı üzerine hızla eşleştirilerek; konumsal hız verilerinden, İBB trafik uygulamalarında kullanılan anlık trafik yoğunluk verileri üretilmektedir.

Anahtar Sözcükler: Coğrafi Konum, GPS, İBB, Trafik, Trafik Yoğunluk Haritası, Yol Ağı Eşleştirme

GİRİŞ

İstanbul kent trafiğinin etkin yönetimi için İstanbul Büyükşehir Belediyesi (İBB) tarafından birçok Akıllı Ulaşım Sistemi çözümü hayata geçirilmektedir. Kent trafiğine ilişkin anlık bilgilerin, alternatif güzergâh önerilerinin, trafik koşullarını etkileyen çalışmaların sunulduğu “İBB CepTrafik [1]”, “İBB Trafik Yoğunluk Haritası [2]”, “İBB Yol Gösteren [3]” uygulamalarında, seyir halindeki araçlardan, İBB mobil trafik uygulama kullanıcılarından (İBB CepTrafik, İBB Yol Gösteren) elde edilen konumsal trafik verileri (GPS verileri), anlık trafik bilgisi üretmek için veri kaynağı olarak kullanılmaktadır. Toplanan konumsal hız verileri, bir dizi ön işlemden geçirilerek, trafik uygulamalarında ve en uygun güzergâhların hesaplanmasında kullanılmak üzere yol ağı ile ilişkilendirilmektedir.

Gerçek zamanlı trafik bilgilerinin kullanıcılara sunulduğu İBB trafik uygulamalarında, trafik yoğunluk verileri dakikada bir güncellenmektedir. Bu nedenle, seyir halindeki araçlardan ve kullanıcılardan toplanan tüm konumsal hız verilerinin, bir dakikadan daha kısa bir sürede, yaklaşık 100m uzunluktaki segmentlerden oluşan İBB yol ağı ile ilişkilendirilmesi gerekmektedir. Aksi takdirde, toplanan konumsal hız verileri güncelliğini kaybetmektedir.

Literatürde konumsal verilerin yol ağı ile eşleştirilmesine yönelik birçok yöntem bulunmasına rağmen, İBB trafik uygulamalarında olduğu gibi, sürekli olarak büyük veri üretildiği durumlarda, yeterli hızda eşleştirme yapılamadığı için performans sorunu ile karşı karşıya kalınmaktadır. Bu çalışmada, İBB trafik

¹ Mustafa Kemal Sürmeneli, İstanbul Metropolitan Municipality, Directorate of Traffic, İstanbul, Turkey, mkemal.surmeneli@ibb.gov.tr

² Esmâ Dilek, İstanbul Metropolitan Municipality, Directorate of Traffic, İstanbul, Turkey, esma.dilek@ibb.gov.tr

uygulamalarından toplanan konumsal (GPS) hız verilerinin yol ağı ile eşleştirilmesi, yani üst üste çakıştırarak hangi yol segmentine ait olduğunu bulmak için kullanılan, mesafe ölçmeye dayalı klasik eşleme yöntemlerinden daha hızlı çalışan ve İBB Ulaşım Daire Başkanlığı mühendisleri tarafından geliştirilen algoritma hakkında bilgi verilmektedir. Bu algoritma, vektörel yol ağının bir grid ile parçalara ayrılması ve herhangi bir GPS verisinin bu parçalardan hangisi üzerinde bulunduğu tespitine dayalı hızlandırılmış bir eşleme yöntemidir.

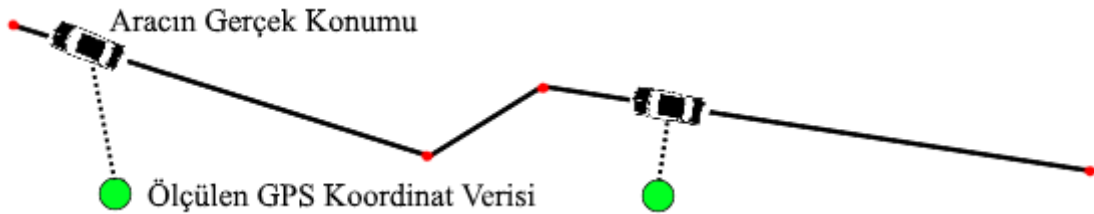
İBB trafik uygulamalarında, toplanan GPS verilerinden trafik yoğunluk haritası elde etme sürecinde, toplanan her GPS verisi ile kullanılan yol ağındaki yaklaşık 100 metre mesafeden oluşan yol segmentleri arasındaki mesafeler hesaplanmaktadır. Herhangi bir GPS verisi ile yol segmenti arasındaki mesafenin 20 metrenin altında ve açıl olarak aynı doğrultuda olduğu durumda, GPS verisinin segment üzerinde olduğu varsayılmaktadır. Bu süreçte, noktaya uzak mesafede olan; eşleşme olasılığı bulunmayan yol segmentlerine olan uzaklıklar da hesaplanmaktadır. Bu çalışmada geliştirilen hızlandırılmış eşleme yönteminde, GPS verisine en yakın olabilecek segmentler diğerlerinden ayırt edilerek; GPS verisinin ait olduğu segmentin hızlıca bulunması ve trafik yoğunluk uygulamalarına entegre edilmesi sağlanmaktadır.

Bu çalışma aşağıda belirtildiği sırada yapılandırılmıştır:

1. Bölümde Yol Ağı İle Eşleştirme;
2. Bölümde Mesafe Ölçüm Yöntemleri;
3. Bölümde Noktasal Verinin Belirli Bir Vektörel Yol Segmentine Olan Euclid Mesafesinin Hesaplanması;
4. Bölümde Yol Eşleme Algoritmasının İBB Trafik Uygulamalarında Kullanımı;
5. Bölümde Yol Ağı Mesafe Ölçümlerinde Optimizasyon: Gridlere Bölme Yöntemi;
6. Bölümde Algoritmanın Performansı hakkında bilgi verilmiştir.
7. Bölümde ise çalışmada elde edilen sonuçlar özetlenmiştir.

1. YOL AĞI İLE EŞLEŞTİRME

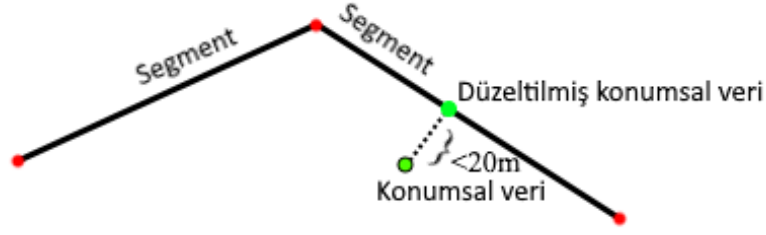
Konumsal veri kaynakları (örnek: GPS verileri), çevresel faktörler nedeniyle sapmaya uğramakta ve her zaman tam ve kesin sonuçlar üretmemektedir. Elde edilen (toplanan) konum verisi ile konum verisi istemcisi olan cihazın gerçekte bulunduğu konum arasında farklılık olabilmektedir. Bu durumda, toplanan konum verisindeki bu hatanın (sapmanın) mümkün olan en yüksek doğrulukta düzeltilmesi gerekmektedir. Bu düzeltme, bir dizi matematiksel işlemi barındırmaktadır. Ham olarak gelen bu konumsal verilerin, gerekli düzeltmelerden sonra harita üzerine işlenmesine harita eşleme işlemi denir [4]. Eğer konumsal veriler, yol ağıyla eşleştirilirse, buna da yol eşleme işlemi denir.



Şekil 1. Gerçek Konum, Toplanan Konum

Düzeltilme işlemleri sonrasında, konumsal bir veri, mevcut vektörel yol ağında, en yakın segment üzerinde bir izdüşüm noktasına dönüştürülmektedir. Burada söz konusu yakınlık, belirli bir tolerans değerinin altındaki yakınlığı ifade etmektedir. Aksi takdirde, yol ile gerçekte ilişkisi olmayan konumsal verilerin yola eşlenmesi söz

konusu olabilmektedir. İBB trafik uygulamalarında, yakınlık tolerans değeri olarak 20 metre tercih edilmiştir. (Bknz. Şekil 2)



Şekil 2. Konumsal Verinin Segment'e İzdüşümü

Şekil 2'de yol ağının her bir vektörel alt birimi olan segmentlere izdüşümü yapılmış; ham koordinat verisi görselleştirilmiştir.

2. MESAFE ÖLÇÜM YÖNTEMLERİ

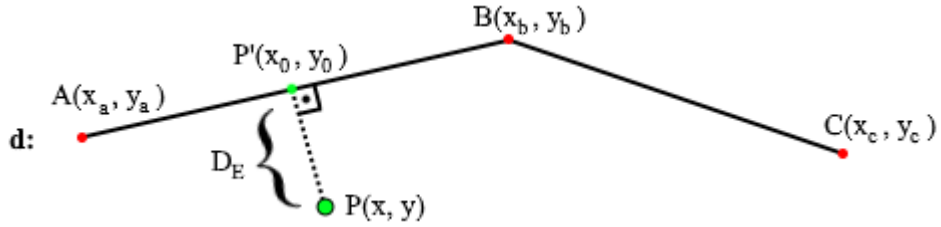
Konumsal verilerin, vektörel yol segmentine olan mesafesini hesaplamada yaygın olarak kullanılan iki yöntem vardır [5][6]:

2.1 Euclid Mesafesi

Şekil 3'te gösterilen bir $P(x, y)$ noktasının d eğrisi (yol ağı) üzerindeki bir $P'(x_0, y_0)$ noktasına olan mesafesine *Euclid mesafesi* denir.

$$D_E = \sqrt{(x - x_0)^2 + (y - y_0)^2} \quad (1)$$

şeklinde hesaplanır.



Şekil 3. Euclid Mesafesi

Şekil 3'te $P'(x_0, y_0)$ noktası $P(x, y)$ noktasının AB doğru parçası (segmenti) üzerindeki izdüşüm noktasıdır.

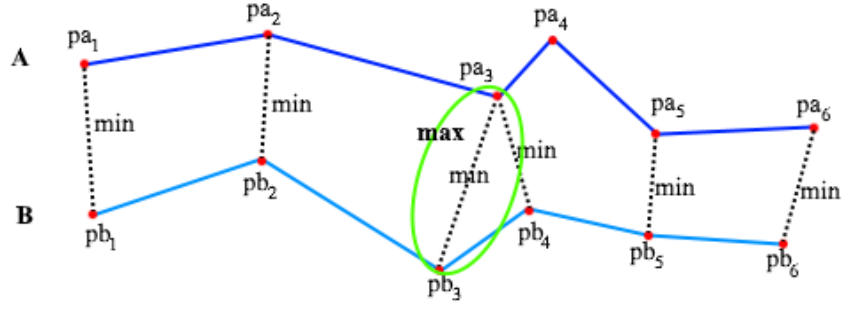
2.2 Hausdorff Mesafesi

$A = pa_1, pa_2, \dots, pa_m$ ve $B = pb_1, pb_2, \dots, pb_n$ noktalarından geçen

yol eğrileri için;

$$H(A, B) = \max(h(A, B), h(B, A)) \quad \text{öyle ki; } h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\| \quad (2)$$

olan $H(A, B)$ değerine A ve B yol eğrileri arasındaki *Hausdorff mesafesi* denir.



Şekil 4. Hausdorff Mesafesi

Şekil 4'e göre A yolu ile B yolu arasındaki Hausdorff mesafesi $\|pa_3 - pb_3\|$ 'dir.

Eğer $\|pa_3 - pb_3\| < 20$ ise (yakınlık tolerans değeri olarak 20 metre kabul edildiğinde) A ve B yol eğrileri üst üste çakıştırılabilir; yani B yolu A yoluna eşlenebilir.

Yukarıda kısaca bahsettiğimiz yöntemlerden Euclid mesafe ölçme yöntemi ile noktanın noktaya ya da noktanın doğruya olan mesafesi; Hausdorff mesafe ölçme yöntemi ile doğrunun doğruya olan mesafesi hesaplanmaktadır. Bu çalışmada, toplanan noktasal GPS verilerinin, vektör tabanlı yol ağının hangi segmentine daha yakın olduğu hesaplandığı için Euclid mesafe ölçme yöntemi kullanılmıştır.

3. NOKTASAL VERİNİN BELİRLİ BİR VEKTÖREL YOL SEGMENTİNE OLAN EUCLİD MESAFESİNİN HESAPLANMASI

Vektörel yol segmentinin başlangıç ve bitiş noktaları $A(x_a, y_a)$ ve $B(x_b, y_b)$; konumsal nokta ise $P(x, y)$ olsun. P noktasının (konumsal koordinat, GPS verisi) AB doğru parçasına (vektörel yol segmenti) olan uzaklığı, noktanın doğruya olan mesafesi hesabından:

$$d = \frac{|(y_a - y_b)x + (x_b - x_a)y + x_a y_b - x_b y_a|}{\sqrt{(y_a - y_b)^2 + (x_b - x_a)^2}} \quad (3)$$

formülüyle bulunur.

Şekil 3'teki $P'(x_0, y_0)$ noktasının hesaplanması ise şu şekilde olur:

$$P'(x_0, y_0) = \left\{ x_0 = \frac{(x_b - x_a)^2 x - (y_a - y_b)(x_b - x_a)y - (y_a - y_b)(x_a y_b - x_b y_a)}{(y_a - y_b)^2 + (x_b - x_a)^2} y_0 \right. \\ \left. = \frac{(y_a - y_b)^2 y - (y_a - y_b)(x_b - x_a)x - (x_b - x_a)(x_a y_b - x_b y_a)}{(y_a - y_b)^2 + (x_b - x_a)^2} \right. \quad (4)$$

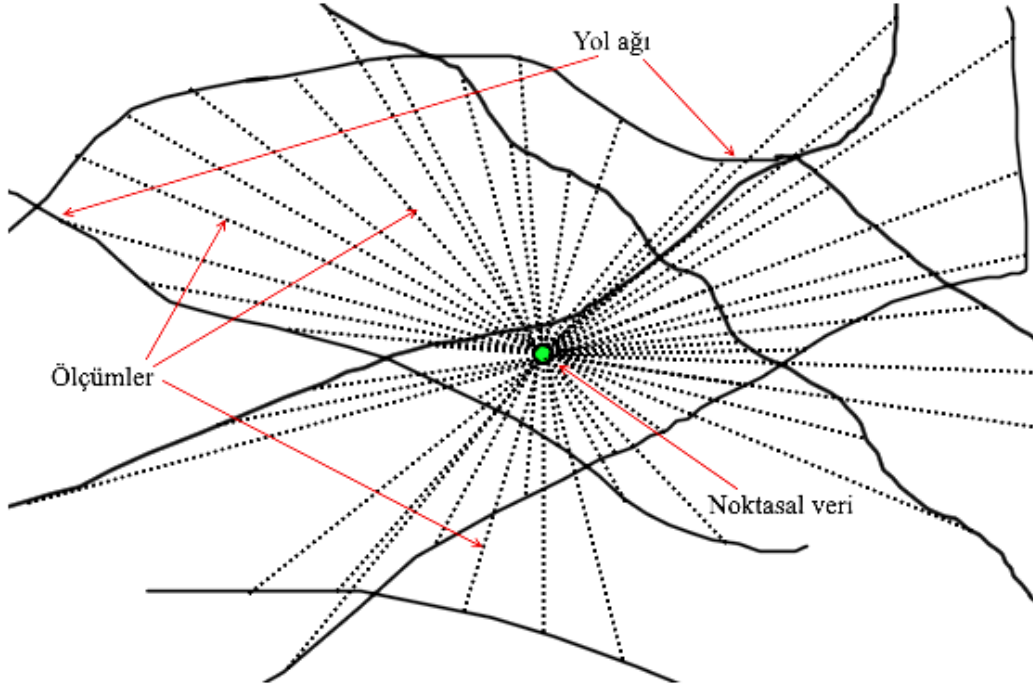
Konumsal bir koordinat verisinin, vektör tabanlı bir yol ağındaki en yakın segment ile eşlenmesi, yukarıdaki denklem (3)'ten faydalanarak uzaklıkların hesaplanması ve bu uzaklıklardan en küçüğünün alınmasıyla sağlanmaktadır.

4. YOL EŞLEME ALGORİTMASININ İBB TRAFİK UYGULANMALARINDA KULLANIMI

Seyir halinde olan araçlardan ve mobil uygulama kullanıcılarından elde edilen konumsal hız verileri, İBB trafik uygulamalarında kullanıcılara sunulan trafik yoğunluk bilgilerinin elde edilmesinde kullanılmaktadır. Mobil

cihazlardan (İBB Yol Gösteren, İBB CepTrafik uygulaması yüklü akıllı telefonlar, vb.) toplanan konumsal koordinat verileri ile uygulamalarda kullanılan vektör tabanlı yol ağının eşlenmesi sonrasında; eşlenen veriler harita tabanlı anlık trafik yoğunluk verilerinin üretilmesinde kullanılmaktadır.

İBB trafik uygulamalarında, toplanan konumsal koordinat verilerinin, yol ağını oluşturan yaklaşık 100m uzunluktaki her bir segmente olan uzaklıkları hesaplanmakta ve hesaplanan uzaklıklar içinde yakınlık tolerans değerinin (20 metre) altında kalan en küçük mesafe hangi segmente ait ise konumsal koordinat verisi o segment ile eşlenmektedir. Toplanan her noktasal konum verisinin, vektörel yol ağındaki hangi segmente eşleneceğinin belirlenmesi; segment sayısı kadar Euclid mesafe hesabı yapılmasını gerektirmektedir (Bknz. Şekil 5). Örneğin, İBB trafik uygulamalarında kullanılan yaklaşık 38 bin vektörel segmentten oluşan bir yol ağında, noktasal verinin hangi segmente 20m'nin altında mesafelerde yakın olduğunun tespiti için denklem (3)'teki hesabın yaklaşık 38 bin kez yapılması gerekmektedir. Günümüz sunucularında bu işlemin birkaç milisaniye sürmesine karşılık; veri sağlayıcılardan toplanan verilerin dakikada yaklaşık 800.000 noktasal koordinat verisini kapsadığı İBB trafik uygulamalarında (örnek: İBB Yol Gösteren); eşleme hesaplamasının tamamlanmasının dakikalar seviyesine çıkması, yoğunluk haritasındaki verilerin gerçek zamanlılık hassasiyetini olumsuz etkilemektedir. Bu nedenle, İBB trafik uygulamalarında olduğu gibi büyük veri üretilen harita tabanlı sistemler için eşleme işleminin hızlı bir şekilde yapılmasına ihtiyaç vardır.

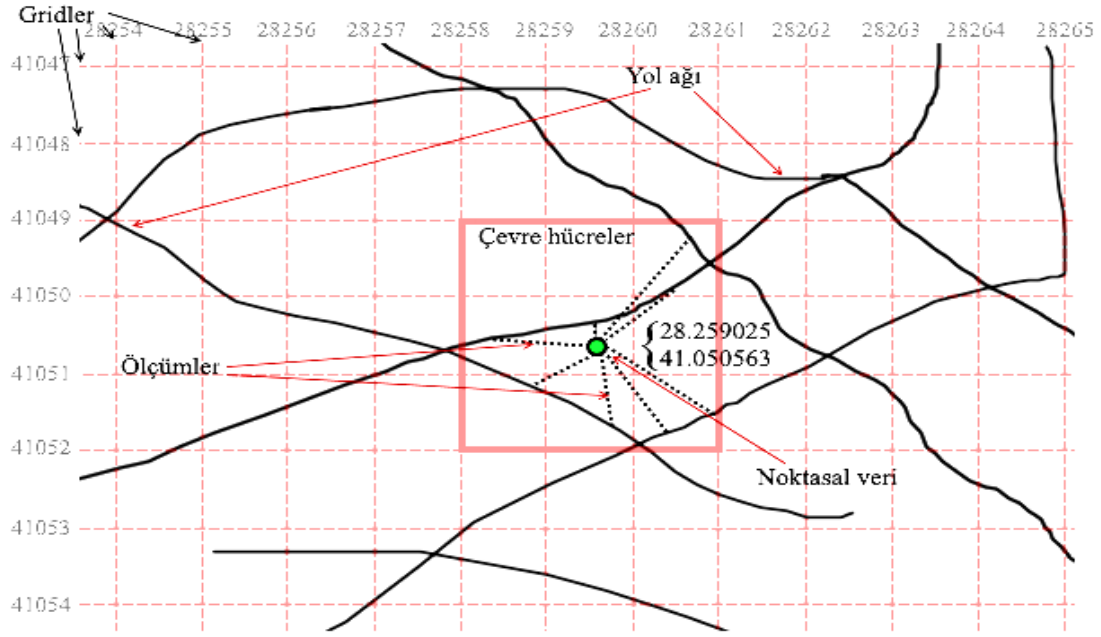


Şekil 5. Konumsal Veri ile Yol Ağını Oluşturan Segmentler Arası Mesafelerin Ölçümleri

Şekil 5'te yol ağına eşlenmesi gereken noktasal bir koordinat verisinin (yeşil nokta), yol ağındaki tüm segmentlere (kalın çizgiler) olan mesafesi (kesikli çizgiler) görselleştirilmiştir.

5. YOL AĞI MESAFE ÖLÇÜMLERİNDE OPTİMİZASYON: GRİDLERE BÖLME YÖNTEMİ

Dakikada bir güncellenen İBB trafik yoğunluk verilerini elde etmek için, konumsal verilerin, yol ağını oluşturan ilgili segment ile eşleştirilmesi sırasındaki işlem zamanının düşürülmesi ve yol ağı eşlemesinde tüm segmentler için ayrı ayrı mesafe ölçüm işleminin neden olduğu hesaplamaların mümkün olduğunca azaltılması gereklidir. Bu çalışmada yapılan analizlerde, konumsal verilerin ait olmadığı segmentlere olan uzaklık hesaplamalarının, tüm hesaplamaların ortalama %99,98'ini oluşturduğu tespit edilmiştir.



Şekil 6. Gridlere Bölme

Şekil 6’da görüldüğü gibi, noktasal koordinat verisinin, tüm segmentlere olan uzaklıklarını hesaplamak yerine; oluşturulan grid (hücre) içerisinde, kendisinin bulunduğu grid ile bu gridin çevresindeki diğer hücrelerin oluşturduğu çevresel grid hücreleri içerisinde kalan segmentler için uzaklık ölçümü yapılması; konumsal bir verinin hangi yol segmenti ile eşleştirileceğini tespit etmek için yeterlidir.

5.1 Yol Ağının Gridlere Bölünmesi

Bu çalışmada, İBB trafik uygulamalarında kullanılan yol ağını kapsayacak bir grid sistemi oluşturulmasında, GPS koordinat sisteminin Latitude ve Longitude bileşenlerinin değerleri, nokta dışında baştan 5 hanesini alarak birleştirilmiş ve grid ID’leri (hücre numaraları) üretilmiştir. Örneğin, yol segmentini oluşturan koordinatlardan biri olan (28.259025 41.050563) koordinatı için (28.259025 41.050563) 2825941050 grid ID değeri üretilmiştir. Bu grid ID, (28.259025 41.050563) noktasının da içinde yer aldığı grid hücresinin ID’sidir.

Grid ID’ler oluştururken, segmentlerimizin ortalama uzunlukları mertebesinde bir hassasiyet oluşturmak amacıyla, koordinat bileşenlerinin noktadan sonra 3 (toplam 5) hanesi kullanılmıştır. Eğer noktadan sonra 2 hane kullanılırsa 1113 m’lik; 3 hane kullanılırsa 111.3 m’lik; 4 hane kullanılırsa 11.13 m’lik bir hassasiyete sahip olunmaktadır (Bknz Tablo 1). İBB yol segmentleri ortalama 100 metre seviyesinde uzunluklara sahip olduğu için noktadan sonra 3 hane hassasiyetli grid ID’ler oluşturulmuştur.

Tablo 1. Koordinatlar için açı-mesafe dönüşümleri [7]

Grid ID Hassasiyeti	Ondalık Derece	Derece	Mesafe
2	1.0	1°0’0"	111.319 km
3	0.1	0°6’0"	11.132 km
4	0.01	0°0’36"	1.113 km
5	0.001	0°0’3.6"	111.3 m
6	0.0001	0°0’0.36"	11.13 m
7	0.00001	0°0’0.036"	1.11 m
8	0.000001	0°0’0.0036"	11.1 cm

Bu çalışmada, İBB trafik uygulamasında kullanılan yol ağını oluşturan her bir segment için, içerdiği nokta sayısı kadar grid ID’si tanımlanmıştır. Toplamda, İstanbul yol ağı için 68 bin’den fazla grid ID’si tanımlanmıştır.

Örneğin, İBB Trafik Yoğunluk Haritasında 170 numaralı segmentin çiziminde kullanılan konumsal veriler şu şekildedir:

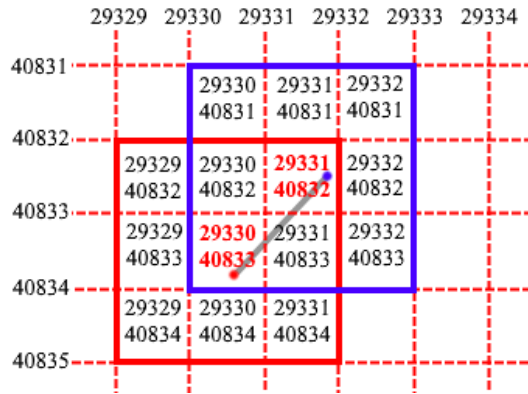
LINESTRING (29.330584093341074 40.833150619995649, 29.331603107849222 40.832691484328954).

Bu LineString’de görüldüğü üzere 170 no’lu segment, iki adet noktasal konum verisi içerdiği için, bu segment aşağıda gösterildiği gibi iki adet grid ID’sine sahiptir:

29.330584093341074 40.833150619995649 → 2933040833

29.331603107849222 40.832691484328954 → 2933140832

Şekil 7’de gösterildiği gibi, bu segmenti oluşturan grid’lerin çevre hücreleri de alındığında, bu segmente yakın olan noktasal koordinatların sınırları belirlenmektedir.



Şekil 7. 170 no’lu Segmentin Çevre Grid Hücreleri

Bu çalışmada, yol ağını oluşturan her bir segmente ait “Çevresel Grid Hücreleri”nin ID’leri, veritabanında bir tabloya kaydedilmiş (Bknz Tablo 2); toplanan konumsal verileri, kullanılan yol ağı ile eşleştirmek için mesafe ölçümü yapılacak segmentlerin listesi bu tablodan çekilmiştir.

Tablo 2. ID’si 170 olan Segment’e ait grid hücre ID’leri listesi

SegGridListID	segID	GridID	NeighborType
2851	170	2933040833	0
2849	170	2933040832	1
2856	170	2933140832	2
2857	170	2933140833	3
2859	170	2933140834	4
2853	170	2933040834	5
2847	170	2932940834	6
2846	170	2932940833	7
2845	170	2932940832	8
2855	170	2933140832	0
2854	170	2933140831	1
2860	170	2933240831	2
2861	170	2933240832	3
2862	170	2933240833	4
2858	170	2933140833	5
2852	170	2933040833	6
2850	170	2933040832	7
2848	170	2933040831	8

Bu

tabloda;

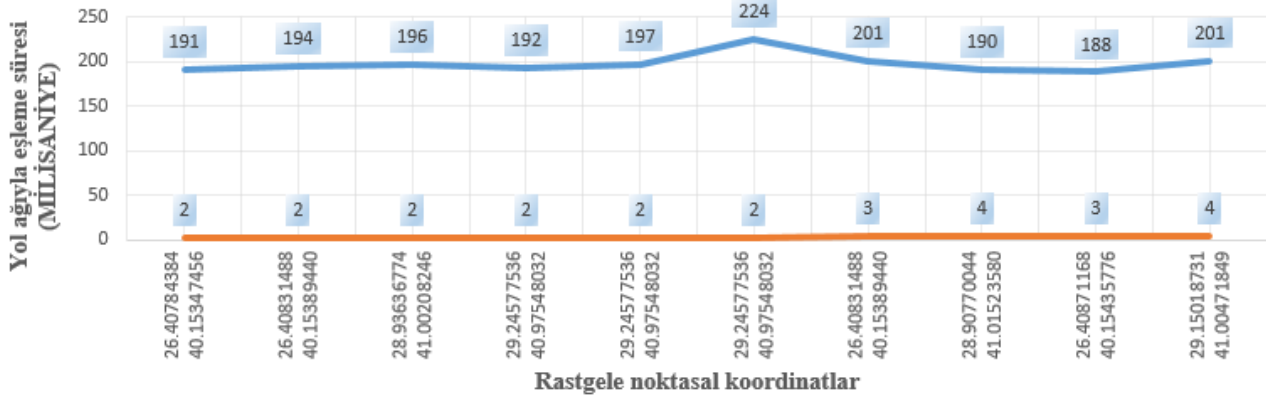
SegGridListID: Tablonun birincil anahtarı
NeighborType: 0 ise segmentin içinde bulunduğu grid(ler)i, 1 ile 8 arası sayılar ise komşu gridleri belirtmektedir. Üst orta grid 1 olacak şekilde, saat yönünde 1'den 8'e kadar numaralar verilmiştir.

Böylece, toplanan konumsal verilerin, uygulamalarda kullanılan vektörel tabanlı yol ağı ile eşlenmesi işlemi, aşağıda belirtilen üç adımdan oluşmaktadır:

1. Noktasal koordinat verisi için Grid ID değeri hesaplanması.
2. Bu Grid ID'nin, SegmentGridList tablosunun GridID alanındaki değerler içinden filtrelenmesi.
3. Filtreleme sonunda elde edilen segID listesinden, mesafe ölçümü yapılacak segment listesinin elde edilmesi.

6. ALGORİTMANIN PERFORMANSI

Çalışma kapsamında yapılan analizlerde, konumsal verilerin yol ağı eşleştirmesinde kullanılan literatürdeki uzaklık hesaplamasına dayalı yöntem ile bu çalışma kapsamında geliştirilen grid sistemine dayalı algoritmanın performans karşılaştırmasıyla, Şekil 9'da görülen grafik elde edilmiştir.



Şekil 9. Konumsal Verinin Yol Ağı ile Eşlenmesi Süreleri (ms)

Şekil 9.'da görüldüğü üzere, rastgele seçilmiş örnek noktasal verilere, klasik yol eşleme ve grid sistemi algoritmasıyla iyileştirilmiş yol eşleme işlemi uygulandığında, 50 ile 100 kat arası bir performans kazancı elde edilmiştir.

7. SONUÇ

Bu çalışmada, sahadan toplanan noktasal verilerin, yol ağına eşleme performansını artıran algoritma hakkında bilgi verilmiştir. Yapılan testlerde, literatürde kullanılan yöntemlerle yaklaşık 350 milisaniye süren eşleme işlemi, 3 milisaniyeye kadar düşürülmüştür. Geliştirilen algoritma ile noktasal verinin eşleştirileceği segmenti bulmak için noktanın çevresinde olamayacak kadar uzaktaki yol segmentleri hesaplama dışı bırakılmış; böylece uzaklık hesaplaması yapılacak segment sayısı minimize edilmiştir.

İBB trafik uygulamalarında, dakikada bir toplanan yaklaşık 800.000 konumsal verinin, geliştirilen algoritma ile eşleştirilmesi, ortalama 20 saniye sürmekte ve uygulamalarda kullanılan verilerin dakikada bir güncellenmesi sağlanmaktadır. Her dakikada büyük veri setlerinin toplandığı benzer sistemler için, geliştirilen grid algoritması, literatürdeki yöntemlere göre yaklaşık 100 kata varan oranlarda performans artışı sağlamaktadır.

Noktasal veriler için geliştirilmiş olan bu yöntem bir sonraki aşamada çizgisel veri setleri için de genişletilebilir. Dikkat edilmesi gereken husus, eşleştirilecek çizgisel verinin, gridlere bölünmüş mevcut yol ağında hangi gridler içerisinde yer aldığı belirlenmesinde uygulanacak yöntemin iyi tasarlanması gerektirir.

KAYNAKLAR

- [1] <http://tkm.ibb.gov.tr/ibb-cep-trafik>
- [2] http://tkm.ibb.gov.tr/YHarita/Harita_tr.aspx
- [3] <http://tkm.ibb.gov.tr/ibbyolgosteren>
- [4] Aktuğ, B., Çelik,R.N., 2008, "Oem Gps Alıcıları ve Bir Harita-Eşleme Yöntemi", HGK Harita Dergisi
- [5] Zhang, M., 2009, "Methods and Implementations of Road-Network Matching"
- [6] Hacı, M., Gökğöz, T., 2016 "An Experiment On Distance Metrics Used For Road Matching In Data Integration", Sigma J Eng & Nat Sci
- [7] https://www.sunearthtools.com/dp/tools/pos_earth.php