

# Kaza-olay algılama algoritmalarının mikroskobik benzetim ile modellenmesi ve değerlendirilmesi: D 100 Karayolu örneği, İstanbul

Doç. Dr. Iğın Gökaşar<sup>1,\*</sup>

<sup>1</sup> Department of Civil Engineering, Faculty of Engineering, Boğaziçi University, İstanbul, Türkiye

\*Sorumlu yazar: [ilgin.gokasar@boun.edu.tr](mailto:ilgin.gokasar@boun.edu.tr)

**Özet:** Üç kaza tespit algoritması (Kaliforniya, Minnesota ve SNS algoritmaları) D 100 Karayolunun 2.5km'lik bir kısmında PARAMICS mikroskobik benzetim yazılımıyla modellenmiştir. PARAMICS modeli ile değerlendirilen üç algoritmanın da İstanbul trafiği için çok uygun olduğu gösterilmiştir. Kaliforniya, Minnesota ve SNS Algoritmaları orta trafik talebinde DR değerlerini sırasıyla %25, %31,5 ve %25 yükselmiştir. MTTD değerleri orta trafik talebinde sırasıyla 48.99%, 59.26% ve 48.80% azalmıştır. Minnesota Algoritması en düşük MTTD (0.88) değerinde çok iyi performans göstermektedir. Bunun yanında, SNS Algoritması düşük trafik düzeyinde sadece 1 yanlış alarm verilmiştir. Tüm test edilmiş algoritmalar, dedektörler arası eşit ve 800m'den az olduğunda Trafik Kontrol Merkezinden alınan gerçek zamanlı trafik verileri kullanılarak uygulanabilir.

**Anahtar Kelimeler:** Kaza-Olay Algılama Algoritmaları, Mikroskobik benzetim, D 100 Karayolu

**Abstract:** Three incident detection algorithms (California, Minnesota and SNS) have been tested on 2.5km-length section of D100 Highway using PARAMICS microscopic simulation software. PARAMICS model evaluation showed that all three algorithms are well suited for the İstanbul traffic. California, Minnesota and SNS Algorithms are better in medium traffic demand since detection rate values are increased by 25%, 31.5% and 25%, respectively. Mean time to detect (MTTD) values are also good in medium traffic demand in that they decrease by 48.99%, 59.26% and 48.80%, respectively. Actually, Minnesota Algorithm had superior performance with the lowest MTTD value (0.88 min). In addition, there is only one false alarm generated by SNS Algorithm in low traffic demand. All of the tested algorithms when properly calibrated can be implemented using real time traffic data provided that distances between the detectors are equally spaced and smaller than 800 m by the traffic control centers.

**Key words:** Incident detection algorithms, Microscopic simulation, D 100 Highway

## 1. Giriş

Ulaşım talebi, metropollerdeki nüfus artışına bağlı olarak yıldan yıla artış göstermektedir. Bu artış sonucunda mevcut yolların kapasitesi aşılarak trafik sıkışıklığı meydana gelmektedir. Ana metropollerde yaşanan trafik sıkışıklığı her yıl üretim kaybına, boşa harcanan yakıt, artan hava kirliliği ve gecikmelere bağlı milyarlarca dolara mal olmaktadır [1].

Trafik sıkışıklığı iki grupta incelenebilir: tekrarlanan ve tekrarlanmayan sıkışıklıklar. İlki belli bir rutine dayalı olarak oluşmakta ve geçici olaylardan etkilenmemektedir. Diğeri ise hava durumu, yol çalışmaları gibi geçici olaylara dayalı olarak oluşmaktadır. Arıza ya da buna benzer engellerin oluşturduğu geçici olaylar trafik sıkışıklığına neden olmakta ve buna da 'kaza-olay' adı verilmektedir. Kaza-olay yönetimi temel bir araç olmakla birlikte; sürücülere karar alma aşamalarında bilgi iletmeyi, birimler arası koordinasyonu arttırmayı, sıkışıklıkları azaltmak veya önlemek amacıyla hızlıca reaksiyon gösterilmesini sağlar. Kaza-olay yönetimi dört aşamadan oluşmaktadır: olay saptaması, müdahale, olay yerinin temizlenmesi ve iyileşme.



Şekil 1. Kaza-olay yönetimi aşamaları [2].

Saptama, olayın meydana gelmesinin belirlenmesidir. Doğrulama ise olayın yeri ve meydana geliş şeklinin belirlenmesidir. Müdahale, devreye girme, koordine olma ve olay yerini temizleyecek olan donanım ve uygun personeli yönetmektir. Olay yerinin temizlenmesi, olaya bağlı oluşan durumların ve olayın sonlanması açısından zamanında ve güvenli şekilde gerçekleşmelidir. Son aşama olan iyileşme ise olaydan etkilenmiş trafiğin eski haline döndürülmesidir [2].

Kaza-olay saptaması trafik yönetiminde yeni bir kavram değildir. Kaza-olay saptaması, 1960'ların ortalarından ve 1970'lerin başından beri standart trafik kaza-olay yönetim

uygulamalarının bir parçası olmuştur. Kaza-olay saptaması, bir olayın varlığını ve konumunu belirleme süreci olan olay saptamasının iki önemli adımı vardır [2]:

1. Bir gözetim sisteminden elde edilen verileri kullanarak tıkanıklık varlığını belirlemek.
2. Tıkanıklığın sebebi bir olay olup olmadığını belirlemek için verileri analiz etmek.

Kaza-olay tespiti, genel olay yönetimi sürecinin önemli bir bileşeni olarak görülebilir. Diğer olay yönetim eylemlerinin uygulanabilmesinden önce bir olayın tespit edilmesi ve doğrulanmış olması gerekir. Kaza-olay tespiti sürecinin başarılı kılmak için, olay olduğu an tespitinin yapılması çok önemlidir. Zamanında ve doğru olay yönetimi, bir olayın mümkün olan en kısa sürede temizlenmemesinin, kaza-olayların olumsuz etkilerini dikkat aldığımız zaman daha önemli olur. Bir olayın saptanmasındaki gecikme uzun araç kuyruklarına ve trafik sıkışıklığına sebep olmaktadır. Bu durumlar da ikincil kaza-olayların temel sebebidir.

## **2. Teori**

Kaza-olay tespit algoritmaları iki kategoriye ayrılabilir: otomatik olan ve olmayan algoritmalar. Otomatik olmayan algoritmalar olayları tespit etmek için sürücülerin raporlarını ve operatörlerin Trafik Kontrol Merkezi'nde bulunan gözetim kameralarındaki gözlemlerini kullanır. Otomatik kaza-olay tespit algoritmaları (AIDA) sensörlerden alınan trafik durum bilgilerinin belirli ön koşulları karşılaması durumunda otomatik olarak bir kaza-olay alarmı verir. Bu çalışmada, Kalifornia, Minesota ve SNS kaza-olay tespit algoritmaları D 100 Karayolunun 2.5 km'lik bir kısmında test edilmiştir.

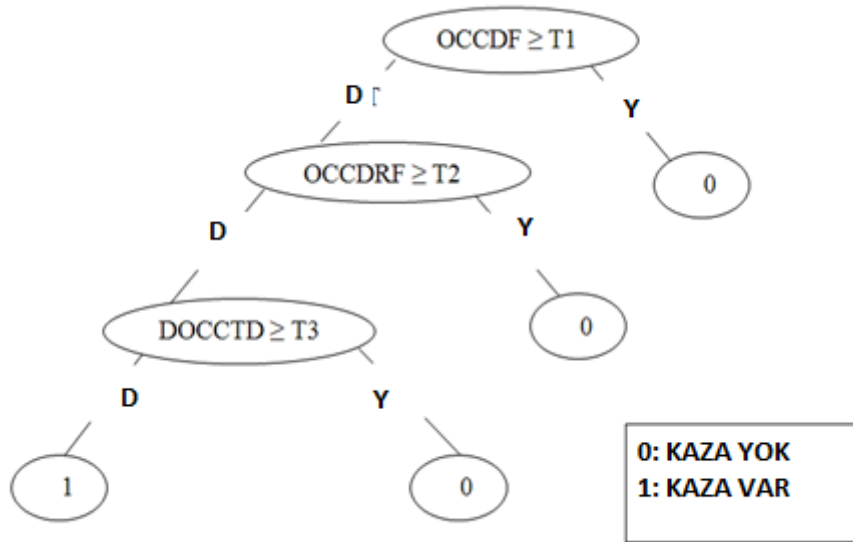
### **2.1. Kaliforniya algoritması**

Kaliforniya Algoritması [3] 1960lı yılların sonlarına doğru Los Angeles Otoyolları Denetim Kontrol Merkezi'ne hizmet etmek amacıyla geliştirilmiştir. Bu algoritma en çok kullanılan otomatik kaza-olay tespit algoritmalarından birisidir. Kaliforniya Algoritmasının yapısı olayların meydana geldiği sırada trafik verilerinde oluşan tipik kalıplar göz önüne alınarak türetilmiştir. Gerçekleşen olay, olay mahalinde otoyolun kapasitesinde azalma yaratır. Eğer sonuçta oluşan kapasite gelen araç hacminden azsa yolda akış yönünün aksine doğru olay yerinden itibaren oluşan bir sıkışıklık meydana gelir. Bir süre sonra, akış yönündeki dedektörler, akış yönünün tersindeki tıkanıklıktan dolayı, trafikte rahatlama olduğunu göstermeye başlar. Bu sebeple, akış yönünün tersindeki sensörler yüksek trafik yoğunluğu gösterirken, akış yönündeki sensörler trafik yoğunluğunu az olarak gösterir.

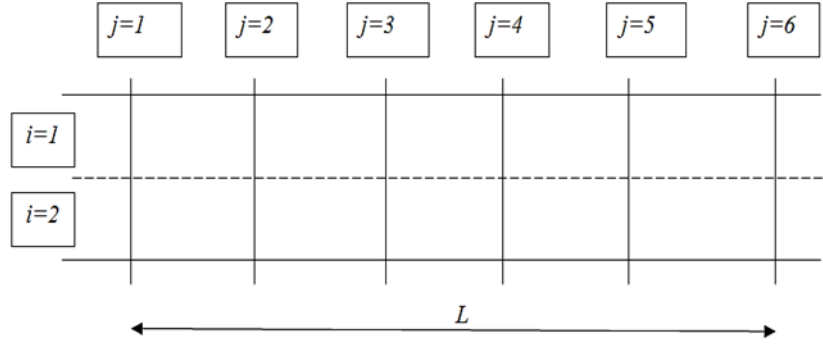
Bu algoritma OCC, DOCC, OCCDF, OCCRF ve DOCCTD adında çeşitli parametreler hesaplar. Bunların her biri bir otoyoldaki belirli bir konumdaki bütün şeritlerin işgal değerlerinin 1-dk aralıklarla yapılan ortalama ölçümlerini kullanır. Tanımları ise aşağıda verildiği gibidir:

- OCC(j,t): t zaman aralığında j istasyonundaki tüm şeritlerde meydana gelen işgal (yüzde),
- DOCC(j,t): t zaman aralığında tüm şeritlerde akış yönünde oluşan işgal j (yüzde),
- OCCDF(j,t): t zaman aralığında tüm şeritlerde meydana gelen j işgallerindeki mekansal farklar (yüzde),
- OCCRF(j,t): t zaman aralığında tüm şeritlerde meydana gelen j işgallerindeki bağlı mekansal farklar (yüzde),
- DOCCTD(j,t): t zaman aralığında tüm şeritlerde meydana gelen j işgallerindeki bağlı geçici fark (yüzde),

Bu algoritma Şekil 2’de gösterilen, en üstteki kök noktasından başlayan üç adımı izler. Bu adımlar alttaki çizimin yardımıyla açıklanmıştır (Şekil 3). Bu çizim, iki şeritli (i), tek trafik akış yönü olan, ve altı ölçüm noktası(j) bulunan bir otoyol kesitidir.



Şekil 2. Kaliforniya Algoritması [4].



**Şekil 3.** İki şeritli yol kesiti.

İlk test uygulanır ve devamında soldaki ( $OCCDF \geq T1$ ) veya sağdaki ( $OCCDF < T1$ ) dal takip edilir. Bu yöntem, sağa doğru giden dal olaysız durumlarla sonuçlanır, bu durumlar bu metotta durum değeri 0 olarak kodlanmıştır. Öbür tarafta, eğer verilen durum doğruysa, sistem OCCRDF'nin ikinci adımına ulaşır.

$$OCCDF(j, t) = OCC(j, t) - OCC(j + 1, t) \quad (2.1)$$

$$OCCRDF(j, t) = \frac{OCC(j, t) - OCC(j + 1, t)}{OCC(j, t)} \quad (2.2)$$

OCCDF ve OCCRDF işgallerdeki farkları ifade eden iki terimdir. Sistemde ikisinin birden yer almasının sebebi, OCCRDF'nin bazı durumlarda düşük trafik rejimlerine yanlış belirtiler göstermesidir. Ancak, azalan işgal değerlerine neden olacak geometrik daralmalar gibi başka sebepler de vardır. Bu yüzden DOCCTD sistemde olaylardan dolayı meydana gelen akış yönündeki ani değişimleri görebilmesi için sisteme yardımcı olur.

$$DOCCTD(j, t) = \frac{OCC(j + 1, t - d) - OCC(j + 1, t)}{OCC(j + 1, t - d)} \quad (2.3)$$

“d” belirtilen zaman aralığını temsil etmektedir. Sistem durum değeri 1'e ulaştığında ki bu üç eşğin de aşıldığını gösterir, olay saptanır. Kaliforniya Algoritmasının matematiksel ifadesi aşağıda gösterildiği gibidir:

$$OCCDF(j, t) = OCC(j, t) - OCC(j + 1, t) \quad (2.4)$$

$$OCCRDF(j, t) = \frac{OCC(j, t) - OCC(j + 1, t)}{OCC(j, t)}$$

$$DOCCTD(j, t) = \frac{OCC(j + 1, t - d) - OCC(j + 1, t)}{OCC(j + 1, t - d)}$$

## 2.2. Minnesota algoritması

Minnesota Algoritması [5] iki ardışık sensör istasyonundaki işgallerin fonksiyonunu değerlendirir. Kaliforniya Algoritmasının aksine, Minnesota Algoritması yanlış alarmlar oluşturacak suni farkları filtreleyebilmek için yumuşatılmış trafik parametrelerini kullanır. Yumuşatılmış işgal oranı akış yönü aksinde bulunan j istasyonunda artıp, akış yönündeki j+1 istasyonunda azaldığı zaman olay saptanır. Yumuşatılmış değerler üssel yumuşatma yöntemi kullanılarak oluşturulur.

Yumuşatılmış değerler ortalama değer veya ölçümlerde gözlenen bir trendi bulmak için kullanılır. Üssel yumuşatma yönteminde, ilk olarak ortalama değer belirlenmiş sayıdaki ölçümler için oluşturulur. Yeni bir ölçüm (occ) elde edildiğinde, denklem 2.5 kullanılarak yeni oluşturulmuş ortalama yaratılır. Bu ortalama kaydedilmiş ölçüm olarak (OCC) kaydedilir. Eski ölçüm (occ) silinir. Devamında, yeni bir OCC oluşturulur.

$$OCC(j, t) = occ(j, t - 1) + \alpha(occ(j, t) - occ(j, t - 1)). \quad (2.5)$$

$\alpha$  Faktörü bu işlem için düzenlenmiş bir yumuşatma faktörüdür. 1 ile 0 arasında bir değeri vardır. Yumuşatılmış değerlerin hesaplanmasından sonra, şu anki durumda oluşan işgal alttaki denklem kullanılarak hesaplanır:

$$\Delta OCC(j, t + k) = OCC(j, t + k) - OCC(j + 1, t + k), \quad (2.6)$$

Burada  $\Delta OCC(j, t+k)$  şu anki durumda('k' aralığına t zamanının ilave edilmesi) j istasyonundaki yumuşatılmış işgal değeri farkıdır,  $OCC(j, t+k)$  şu anki durumda('k' aralığına t zamanının ilave edilmesi) j istasyonundaki yumuşatılmış işgal değeridir. Bu algoritma tekrar eden sıkışıklıkla oluşan olayı ayırt etmek için, geçmiş dönem işgal değerlerindeki farkları kullanır. Bu farklar aşağıdaki denklemle hesaplanır:

$$\Delta OCC(t - n) = OCC(j, t - n) - OCC(j + 1, t - n) \quad (2.7)$$

Burada  $\Delta OCC(t-n)$  şu anki yumuşatılmış işgal durumuyla 'n' zaman aralığı önceki yumuşatılmış işgal durumunun farkıdır.  $OCC(j, t-n)$  (t-n) zaman aralığındaki yumuşatılmış işgal oranıdır.

Aşağıdaki test mekânsal işgalin normalize edilmiş değerlerinin şu anki zaman dilimiyle kritik eşik değerinin( $T_c$ ) arasındaki farkı karşılaştırmaktadır:

$$\frac{\Delta OCC(t)}{\max OCC(t)} > T_c \quad (2.8)$$

$\max OCC(t) = \max [ OCC(j,t-n), OCC(j+1,t-n) ]$  geçmiş ve şu anki zaman dilimindeki trafik durumuyla eşik değerinin arasındaki normalize edilmiş fark tekrar eden bir sıkışmaya bağlı bir olayın varlığını tespit etmek için kullanılır.

$$\frac{\Delta OCC(t+k) - \Delta OCC(t-n)}{\max OCC(t)} > T_i \quad (2.9)$$

Kaza-olay önceden belirlenmiş  $T_i$  ve  $T_c$  eşik değerleri aşıldığında fark edilir. Minnesota Algoritmasının matematiksel gösterimi aşağıdaki gibidir.

$$\begin{aligned} OCC(j,t) &= occ(j,t-1) + \alpha(occ(j,t) - occ(j,t-1)) \\ \Delta OCC(j,t+k) &= OCC(j,t+k) - OCC(j+1,t+k) \\ \Delta OCC(j,t-n) &= OCC(j,t-n) - OCC(j+1,t-n) \\ \max OCC(t) &= \max [ OCC(j,t-n), OCC(j+1,t-n) ] \end{aligned} \quad (2.10)$$

$$\begin{aligned} \frac{\Delta OCC(t)}{\max OCC(t)} &> T_c \\ \frac{\Delta OCC(t) - \Delta OCC(t-d)}{\max OCC(t)} &> T_i \end{aligned}$$

### 2.3. Standart normal sapma (SNS) algoritması

SNS Algoritması önceden belirlenmiş kontrol parametrelerinin trendlerini inceler. Trendlerdeki ani dalgalanmalar eşik değerini aştığı zaman olayın varlığı tespit edilir [5]. Bir olayın olması ile tekrar eden sıklığı ayırt etmek için algoritma toplam değişim yerine kontrol parametrelerindeki değişim oranını inceler. SNS'nin oluşturulması aşağıdaki gibidir:

$$SNS = \frac{x(j,t) - \bar{x}(j,t)}{s} \quad (2.11)$$

Burada, " $x(j,t)$ " t zaman aralığındaki j istasyonundaki trafik kontrol parametresi, " $\bar{x}(j,t)$ " t zaman aralığındaki j istasyonundaki ortalama trafik kontrol parametresi, " $s$ " t zaman aralığındaki trafik kontrol parametresinin standart sapmasıdır.

Bu çalışmada trafik kontrol parametresi işgal olarak seçilmiştir. Gözlemlenmesi beklenen trafik akışı dalgalanması sayısı geçmişte normal bir şekilde dağılmış olan trafik verilerinden hesaplanmaktadır. Olay hesaplanan SNS değeri eşik değeri TSNS yi geçtiği zaman algılanır. SNS'nin matematiksel gösterimi aşağıdaki gibidir:

$$SNS = \frac{x(t) - \bar{x}(t)}{s} > T_{SNS} \quad (2.12)$$

#### 2.4. Başarım ölçütleri

Bu çalışmada, olay belirleme algoritmalarının performanslarını hesaplamak için üç çeşit geçerlilik ölçüsü kullanılmaktadır [6]. Bunlar:

Belirleme Oranı (DR): Belirlenmiş olayın toplam olaya oranını gösterir.

$$DR = \frac{\text{belirlenmiş olay sayısı}}{\text{toplam olay sayısı}}$$

Yanlış Alarm Oranı(FAR): Belirlemenin yeri, olay yeri ile eşleşmediğinde, yanlış alarm kaydedilir. YAO;

$$FAR = \frac{\text{yanlış belirlenmiş olay sayısı}}{\text{toplam olay sayısı}}$$

Belirleme Ortalama Zamanı (MTTD): Olay belirlemesi ile olayın meydana gelme zamanı arasındaki gerçek zaman arasındaki tamamlanan zamandır. Eşik değerlerinin aşıldığı andaki zamanı olay belirleme zamanı olarak kaydedilir.

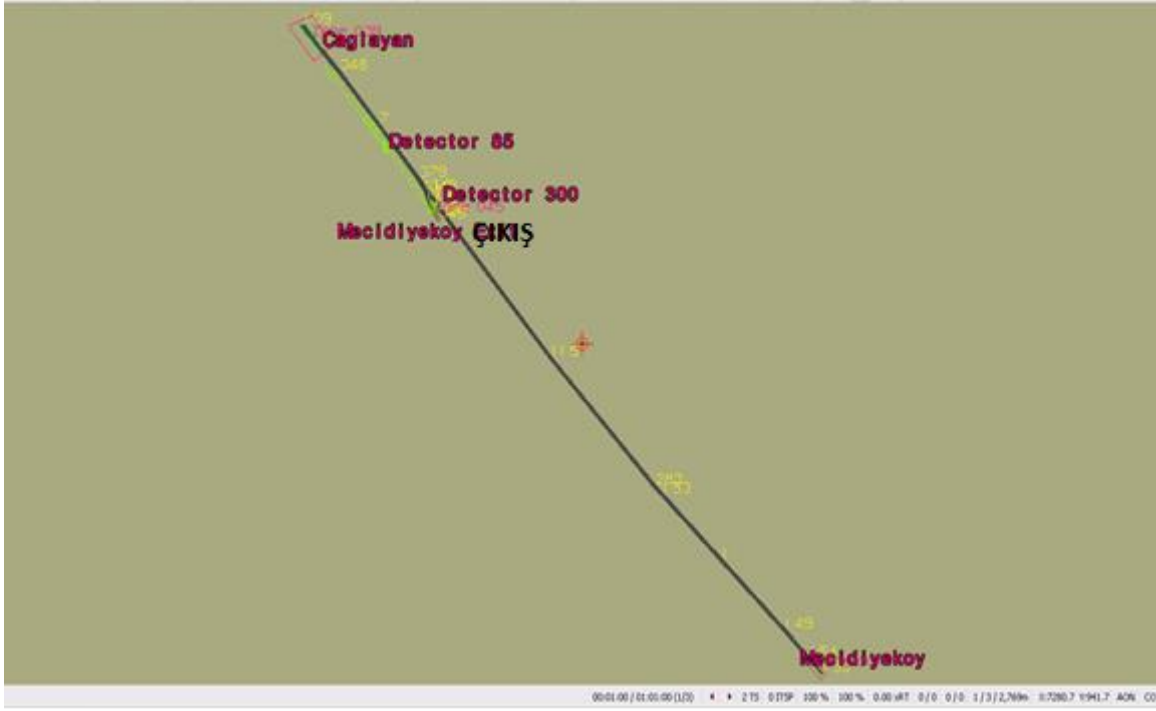
$$MTTD = \frac{1}{n} \sum_{i=1}^n (T_{id} - T_{io}),$$

- n= Benzetim adım sayısı
- $T_{id}$ = Olay belirleme zamanı
- $T_{io}$ = Olay gerçekleşme zamanı

### 3. D100 Otoyol Ağı Benzetimi

Bu çalışmada, D 100 Karayolunun 2.5-km'lik bir bölümü PARAMICS kullanılarak modellenmiştir. Bölüm Çağlayan geçidinin bitişinden başlar ve Mecidiyeköy girişinden önce biter (Şekil 4).

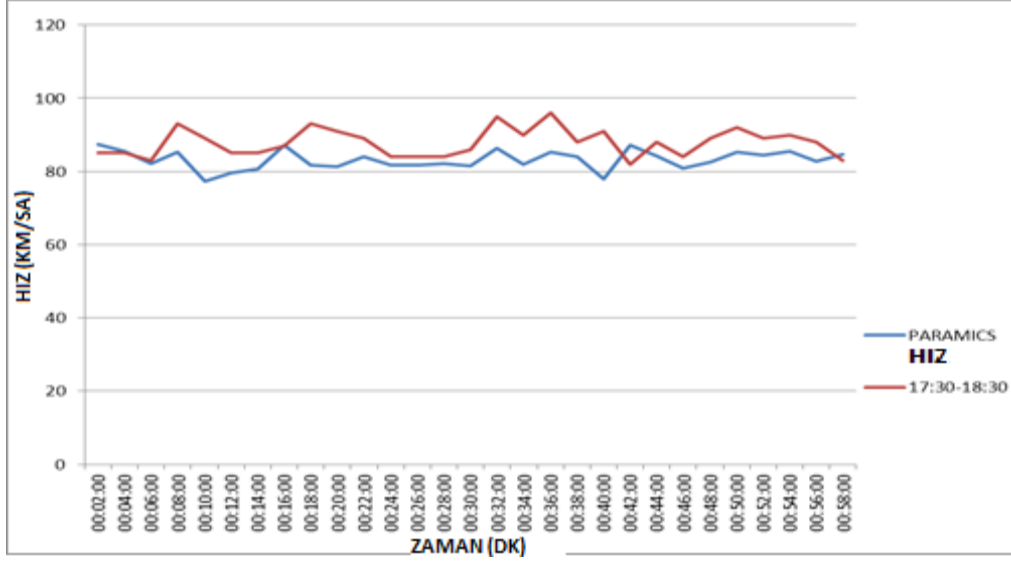




*Şekil 4. Çalışma ağının PARAMICS modeli.*

Modelin tüm davranışı ortalama araç aralığı ve ortalama reaksiyon zamanları ayarlanarak belirlenir. Ortalama araç aralığı zamanı global ortalama taşıt aralığı zamanıdır, saniye biriminden taşıt ile önündeki taşıt arası süredir. Ortalama sürücü reaksiyon zamanı önceki aracın takip ettiği araçlardaki hız değişikliğine gösterdiği tepkideki saniye cinsinden gecikmedir. Değişime gösterdiği reaksiyon [7] iki limit için de alt limit olan 0.5 saniyedir. Üst limit ise ortalama araç aralığı zamanı için 5 ve ortalama sürücü reaksiyon zamanı için 2.2 saniyedir (Mayıs, 1990). Ortalama hedef taşıt aralığı ve ortalama reaksiyon zamanı 1.11 ve 0.89 saniye olarak ayarlanmıştır.

PARAMICS modelde, sensörler gerçek ağdaki trafik hesaplamalarının gerçek ağdaki tam lokasyonlarına konumlandırılmıştır. Gerçek trafik hesaplamaları 2-dakika sensör sayımı, hızı ve doluluk ölçümleri ile karşılaştırılmıştır. Örneğin, Dedektör 18 Ataturk Köprüsü ve Okmeydani geçiti arasına konumlandırılmıştır. Şekil 5'te Dedektör 18 gerçek ve PARAMICS model zamana bağlı hız hesaplamalarını göstermektedir. Çalışma sahasında, akşam en yoğun zaman 16:30 ve 18:30 arasındadır. Çalışma sahası 17:30 to 18:30 arasında 1 saatlik dilimde modellenmiştir. Kalibre edilmiş olan model çalışma sahasındaki olay tespit algoritmalarının benzetimi için kullanıldı.



*Şekil 5. Zamana bağlı hız değerleri, Dedektör 18.*

Frekans, yer, süre, oluş zamanı, geçiş hızı ve karşılıklı araçların hızı PARAMICS olay dosyasında tanımlanabilir. Kaza-olay süresi her senaryo için 20 dakika olarak seçilmiştir. Ozbay et. al. (1999) 20 dakika olay süresinin olay tipi belli olmadığında uygun olduğunu önermiştir. Kaza-olay oluş zamanları benzetimin 10. dakikası ile 30. dakikası arasında değişmektedir.

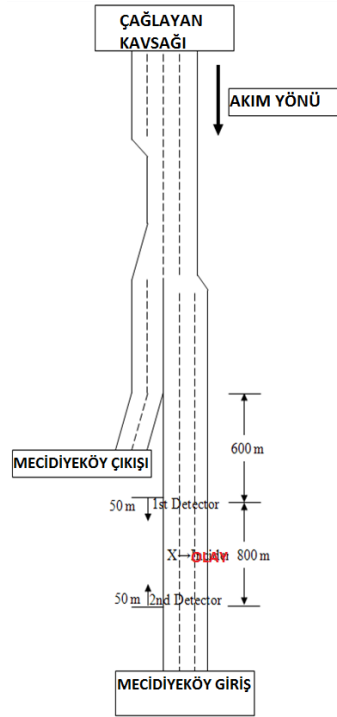
Algoritmalar 2 bitişik istasyondan alınan ölçümlerin karşılaştırılmasına bağlıdır. Verimli bir akıllı ulaşım sistemi stratejisi uygulaması için; dedektörler arası mesafenin standart ve daha önceden planlanmış olması gerekmektedir. Texas Ulaşım Enstitüsü (TTI) olay tespit algoritmalarını değişken dedektör mesafesi ve Houston Gulf Otoyolu (I-45) 1974, deki trafik talep seviyeleri ile analiz etmiştir. Aynı çalışmada, olay tespit algoritmaları en iyi sonuçları bitişik dedektörler arası 160m düşük trafik talebi ve 800m orta trafik talebi olduğunda vermiştir.

Gerçek ağda, modellenmiş bölümler üzerindeki sensörler D100 Karayoluna düzensiz aralıklarla yerleştirilmiştir. Sensörler arası mesafe 535 metreden 2330 metreye kadar farklılık göstermektedir. Algoritmalar 25 olay senaryosuyla düşük ve orta trafik talebi altında test edilmiştir. Testler birbirinden 800m uzak iki hayali dedektör kullanılarak PARAMICS'te modellenmiştir. Kaza-olay parametrelerinin PARAMICS modelindeki değerleri Tablo 1'de özetlenmiştir.

**Tablo 1.** Kaza-olay parametrelerin PARAMICS modelindeki deęerleri.

Süre	20dakika (sabit)
Oluş zamanı	Simulasyonun 10cu ve 30cu dakikası arasında rastgele bir deęer
Location	İki bitişik dedektörün 50 m – 750 m leri içinde rastgele bir deęer
Şerit	1,2 veya 3 (1:acil şerit, 2:orta şerit, 3:refüj şeridi)

Tablo 1’de belirtilen, başlangıç zamanı ve algoritma performanslarının deęerlendirilmesi için lokasyonların dâhil olduęu, sınırlar içerisinde kaza-olay senaryoları MATLAB ile oluşturulmuştur. PARAMICS çalışma modelinin son 2.5km lik bölümünün taslağı Şekil 6’da gösterilmiştir.



**Şekil 6.** PARAMICS çalışma modelinin son 2.5km lik bölüm taslağı.

Önceki çalışmalarda, yoğun trafik durumlarında yanlış alarm oranındaki artış yaşanması nedeniyle algoritmalar zayıf ve orta trafik talebinde deęerlendirilmiştir [3].

Tablo 2, Kaliforniya, Minesota ve SNS Algoritmaları için ayarlanmış parametere deęerlerini göstermektedir.

**Tablo 2.** Kaliforniya, Minesota ve SNS Algoritmaları için ayarlanmış parametere değerleri.

Algoritmanın adı	Düşük trafik talebi altında parametreler	Orta trafik talebi altında parametreler
Kaliforniya Algorithm	T1=-0.20, T2=-1, T3=-1 (Alt sınır)	T1=-0.20 T2=-1, T3=-1 (Alt sınır)
Minesota Algorithm	Tc=-2.5, Ti=-2 (Alt sınır) k=5 min (Güncel period aralığı) n=3 min (Geçmiş period aralığı)	Tc=-2.5, Ti=-2 (Alt sınır) k=5 min (Güncel period aralığı) n=3 min (Geçmiş period aralığı)
SNS Algorithm	Ts=6.5 (Alt sınır) d_time_interval= 5 min (Seçme periyodu)	Ts=6.5 (Alt sınır) d_time_interval= 5 min (Seçme periyodu)

#### 4. Benzetim sonuçları

Kaliforniya, Minesota and SNS Algoritmaları 25 rastgele olay senaryosu kullanılarak, düşük ve orta trafik yoğunluğu altında 1 saatlik benzetim sırasında test edilmiştir. Olayların oluş zamanları (10. ve 30. dakikalar arası), lokasyonları ( 800 m uzaklıkta iki hayali dedektör arası) ve şeritleri (sol, orta, sağ ) bu senaryolarda farklılık göstermektedir. Tespit oranı (DR), yanlış alarm oranı (FAR) ve tespit için ortalama zaman (MTTD) algoritmaların değerlendirilmesinde başarımlar ölçütü olarak kullanılmıştır. Benzetim sonuçları Tablo 3, 4 ve 5'te gösterilmiştir. "Tespit" kolonundaki '1' değeri olayın algoritma tarafından tespit edildiğini göstermektedir.

**Tablo 3.** Kaliforniya, Minnesota ve SNS Algoritmalarının orta trafik talebi altında başarı ölçütleri.

Simulasyon	Olay Parametreleri			Kaliforniya Algoritması		Minnesota Algoritması		SNS Algoritması	
	Oluş Zamanı (dk)	Yer (m)	Şerit No.	Tespit (0:Evet, 1:Hayır)	Tespit Zamanı (dk)	Tespit (0:Evet, 1:Hayır)	Tespit Zamanı (dk)	Tespit (0:Evet, 1:Hayır)	Tespit Zamanı (dk)
1	15	537	2	1	2	1	1	1	2
2	15	490	2	1	2	1	1	1	2
3	14	221	3	1	1	1	1	1	1
4	16	522	1	1	1	1	1	1	3
5	12	224	2	1	3	1	0	1	1
6	13	355	3	1	1	1	0	1	1
7	23	453	3	1	2	1	0	0	-
8	24	470	2	1	3	1	0	0	-
9	24	804	1	1	5	1	4	0	-
10	14	709	1	1	5	1	3	1	4
11	20	428	3	1	2	1	0	1	2
12	16	622	1	1	5	1	2	0	-
13	19	577	2	1	3	1	0	0	-
14	12	691	3	1	5	1	2	1	2
15	24	242	1	1	1	1	0	1	1
16	12	447	1	1	2	1	1	1	1
17	24	522	3	1	3	1	1	1	3
18	13	239	2	1	1	1	0	1	1
19	13	250	1	1	1	1	0	1	1
20	14	373	1	1	1	1	0	1	1
21	17	326	1	1	1	1	0	1	1
22	28	540	3	1	3	1	1	1	2
23	14	205	1	1	1	1	0	1	1
24	29	541	3	1	2	1	1	1	2
25	17	586	1	1	4	1	3	1	2
				DR=2 5	MTTD=2. 4	DR=2 5	MTTD=0.8 8	DR=2 0	MTTD=1. 7

**Tablo 4.** Kaliforniya, Minnesota ve SNS Algoritmalarının düşük trafik talebi altında başarı ölçütleri.

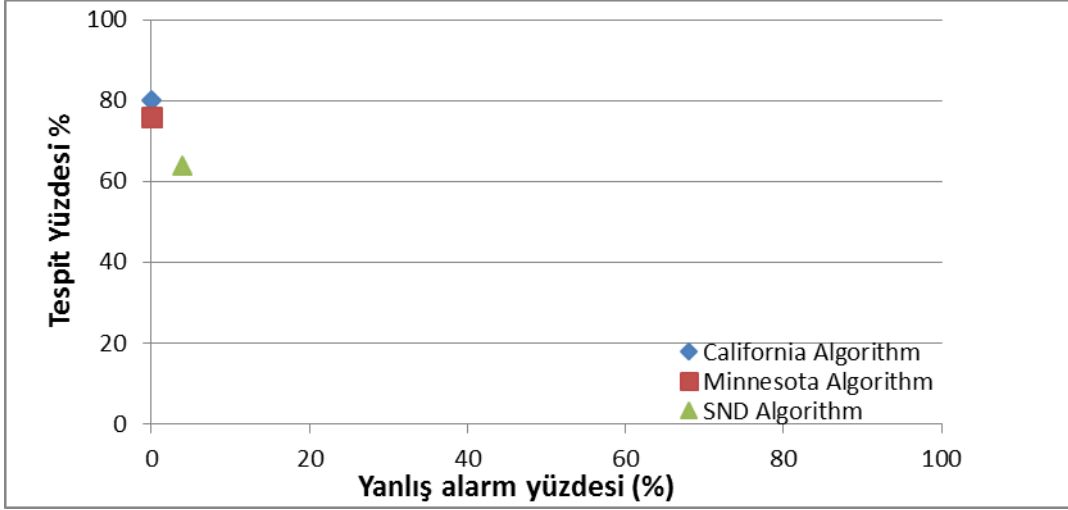
Simulasyon	Olay Parametreleri			Kaliforniya Algoritması		Minnesota Algoritması		SNS Algoritması	
	Oluş Zamanı (dk)	Yer (m)	Şerit No.	Tespit (0:Evet, 1:Hayır)	Tespit Zamanı (dk)	Tespit (0:Evet, 1:Hayır)	Tespit Zamanı (dk)	Tespit (0:Evet, 1:Hayır)	Tespit Zamanı (dk)
1	20	847	1	0	-	0	-	0	-
2	17	194	1	1	1	1	0	1	1
3	16	504	1	1	15	1	13	0	-
4	26	213	2	1	1	1	0	1	1
5	12	784	3	1	6	1	3	1	6
6	21	729	1	0	-	0	-	0	-
7	17	673	1	0	-	0	-	0	-
8	11	618	1	1	8	1	2	1	4
9	23	661	2	1	8	1	1	0	
10	25	352	3	1	3	1	0	1	2
11	24	428	2	1	4	1	0	1	3
12	12	387	3	1	3	1	0	1	2
13	23	224	3	1	2	1	0	1	3
14	13	490	2	1	3	1	0	1	2
15	28	665	3	1	9	0	-	0	-
16	12	212	1	1	1	1	0	0	-
17	26	629	3	1	9	1	8	0	-
18	13	228	3	1	1	1	1	1	1
19	13	381	2	1	1	1	1	1	2
20	24	559	3	1	7	1	4	1	7
21	25	665	1	0	-	0	-	1	9
22	30	211	3	1	1	1	0	1	1
23	18	630	2	1	5	1	5	1	6
24	22	408	3	1	6	1	3	1	3
25	15	691	1	0	-	0	-	0	
				DR=2 0	MTTD=4. 7	DR=1 9	MTTD=2.1 6	DR=1 6	MTTD=3.3 1

*Tablo 5. Benzetim sonuçlarının özeti.*

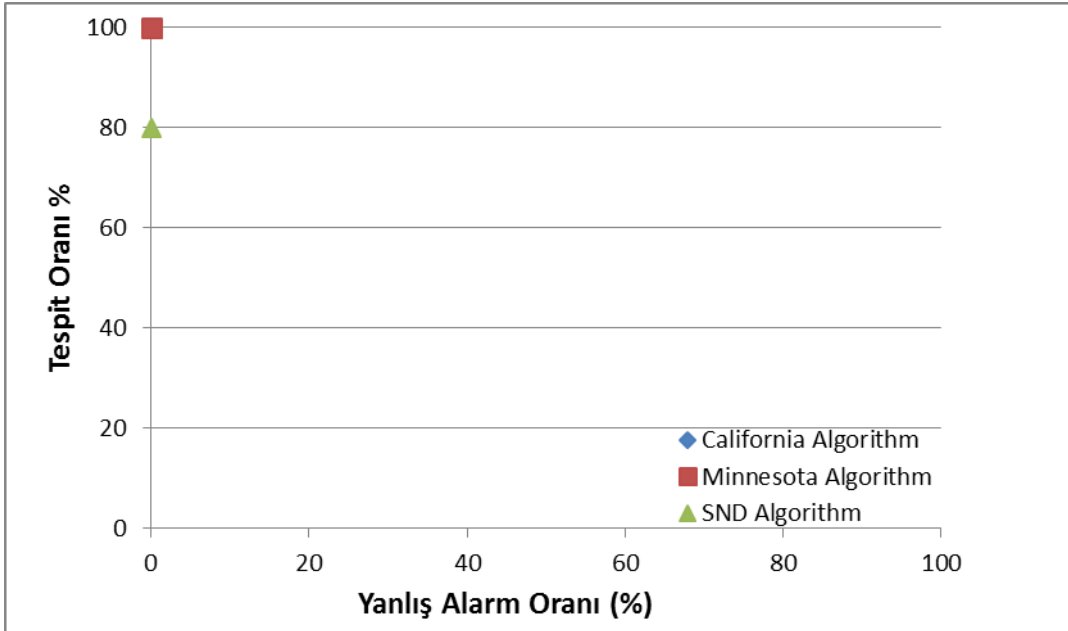
<b>Algoritma adı</b>	<b>Akış Oranı</b>	<b>Simuasyon sayısı</b>	<b>DR</b>	<b>%</b>	<b>FAR</b>	<b>%</b>	<b>MTTD (dk)</b>
Kaliforniya Algoritması	Düşük	25	20	80	0	0	4.70
	Orta	25	25	100	0	0	2.40
	% Değişim		25		0		-48.94
Minnesota Algoritması	Düşük	25	19	76	0	0	2.16
	Orta	25	25	100	0	0	0.88
	% Değişim		31.5		0		-59.26
SNS Algoritması	Düşük	25	16	56	1	4	3.32
	Orta	25	20	80	0	0	1.70
	% Değişim		25		-100		-48.80

Tablo 5 DR değerlerinin düşük orta trafik talebi altında yükselme eğilimde olduğunu göstermektedir. Trafik akışı düşük talep altında düzensizdir fakat orta trafik talebi altında daha stabildir. Bu nedenle, sonralardan olayın etkisi daha şiddetlidir. Fakat ağın içindeki araç sayısı arttığında, hızlı bir şekilde kaza-olay nedeniyle kuyruklar oluşturmaktadırlar. Bu nedenle yukarı akışta yerleştirilen sensörler olayı daha kısa bir zaman diliminde tespit edebilmektedir. Bunun sonucu olarak, MTTD orta trafik talebi altında düşmektedir. Örneğin, MTTD Kaliforniya Algoritması için 4.70 dakikadan 2.40 dakikaya, Minnesota Algoritması için 2.16 da 0.88 dakikaya düşmüştür.

Şekil 7 ve 8 algoritmalarının düşük ve orta trafik talebi altında DR ve FAR değerlerini göstermektedir. DR düşük trafik talebi altında artmaktadır, çünkü trafik akışı daha az düzensizdir ve bu nedenle olayın etkileri sensörler tarafından daha iyi şekilde ayırt edilebilmektedir. DR ve FAR değerleri Minnesota ve Kaliforniya Algoritmaları için orta trafik talebi altında aynıdır, bu nedenle iki algoritmayı göstermek için Şekil 8’de sonuçların üst üste binmesinden ötürü tek nokta bulunmaktadır.



*Şekil 7. Her algoritmanın düşük trafik talebinde DR vs FAR değerleri.*



*Şekil 8. Her algoritmanın orta trafik talebinde DR vs FAR değerleri.*

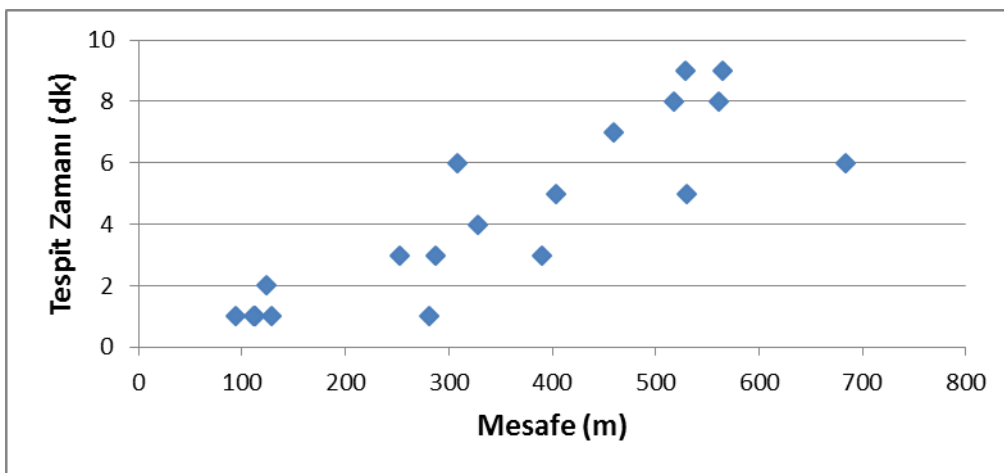
Önceki çalışmalarda, olay doğrulama parametrelerinin artmasıyla DR ve MTTD değerlerinin arttığı ve FAR ın düştüğü belirtilmiştir [6]. Algoritma tarafından 1 fazla tanımlama basamağı gerektiği zaman, olayı tanımlamak biraz mümkündür. Minesota Algoritmasının 2 parametresi vardır ve ortalama en yüksek DR değerine ve en düşük FAR değerine sahiptir. Kaliforniya Algoritmasının 3 farklı tespit parametresi vardır.

Fakat, DR değeri Minesota Algoritması kadar yüksek değildir. Aynı çalışmada, algoritmanın bir olayı (MTTD) birden fazla tespit parametre kullanarak tespit edebilmesi daha fazla vakit almaktadır. Dolayısı ile, MTTD değeri Kaliforniya Algoritmasında Minesota Algoritmasına

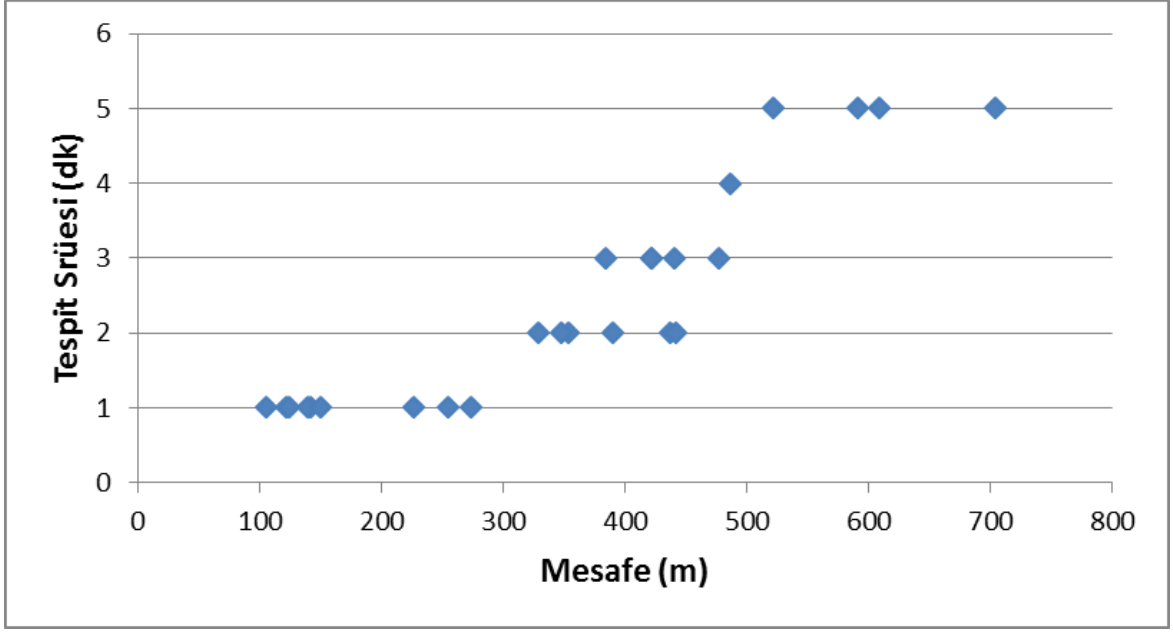


göre daha büyüktür. SNS Algoritmasında sadece 1 değer vardır. Orta ve Düşük trafik talebinde en düşük DR değerine sahiptir.

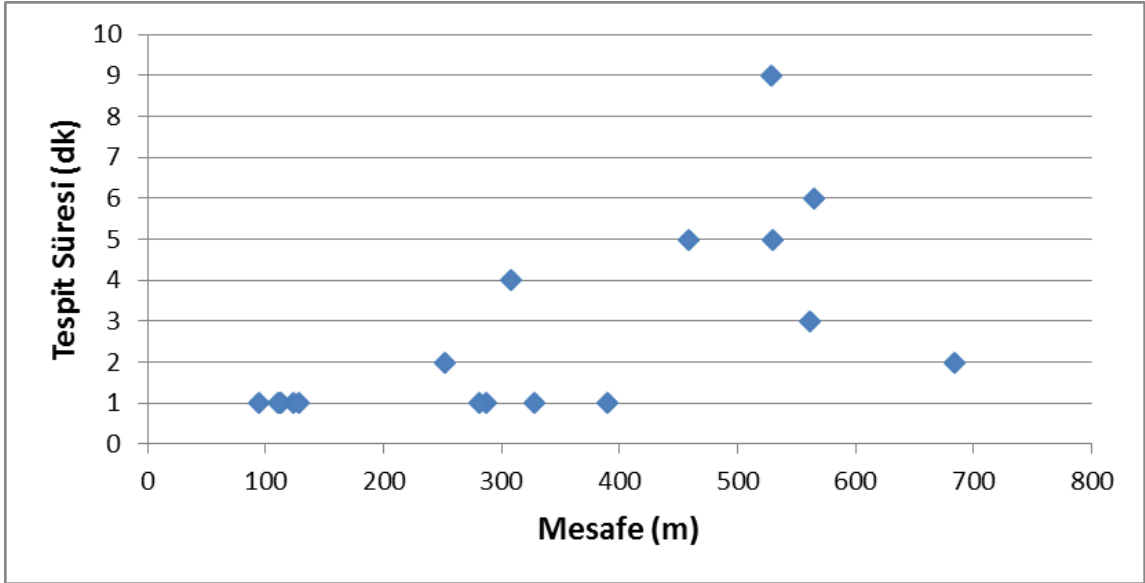
Şekil 9-14 her test edilmiş algoritma için yukarı akış sensörü tarafından değişik olay lokasyonları altındaki tespit zaman değerlerini göstermektedir. Tespit süresi yukarı akış sensöründen olan mesafe arttıkça artmaktadır. Bu etki orta trafik talep senaryolarında açıkça görülmektedir. Sensörden olan mesafe 100m ile 300m arasında olduğunda Kaliforniya Algoritması olayları 1 dakika içinde tespit edebilmektedir. Fakat bu mesafe 500 m veya daha fazlaya çıktığında, tespit süresi 5 dakikaya kadar çıkabilmektedir. 5 dakika tespit süresi çok fazla bir seviyedir ve başarısız olarak gösterilmektedir [5]. Tablo 3 ve 4'te, görülmektedir ki oluş zamanı ve oluş şeridinin tespit performansında çok da fazla bir etkisi yoktur. Diğer yandan, olayın yukarıakış sensöründen olan uzaklığının algoritmanın MTTD değerleri üzerinde ciddi bir etkisi vardır. Mesafe arttıkça MTTD değerleri de artmaktadır. Düşük trafik koşullarında MTTD değerlerindeki artış trafik akışındaki değişiklik nedeniyle çok açık değildir. Fakat orta düzey trafik de olayın lokasyonunu etkisi açıkça görülmektedir. Olay tespit zamanı bazı senaryolar için Minnesota Algoritmasında 0'dır. Geçmiş ve güncel trafik durumlarının karşılaştırılması algoritmanın hızlı bir şekilde tespit etmesini sağlamaktadır. Bu nedenle, tespit zamanı değerleri bu algoritmada daha düşüktür. Şekil 13 ve 14 SNS Algoritması tespit değerlerini farklı olay lokasyonları için göstermektedir. Olay yeri tüm uygulanan algoritmaları benzer şekilde etkilemektedir. Bir olay yukarıakış dedektörüne yakın yerde olduğunda, olay-indirgenmiş sıra olmaktadır. Dolayısı ile dedektör tarafından daha çabuk tespit edilebilmektedir. 600m den büyük mesafeler özellikle düşük trafikte tespit edilebilir değillerdir.



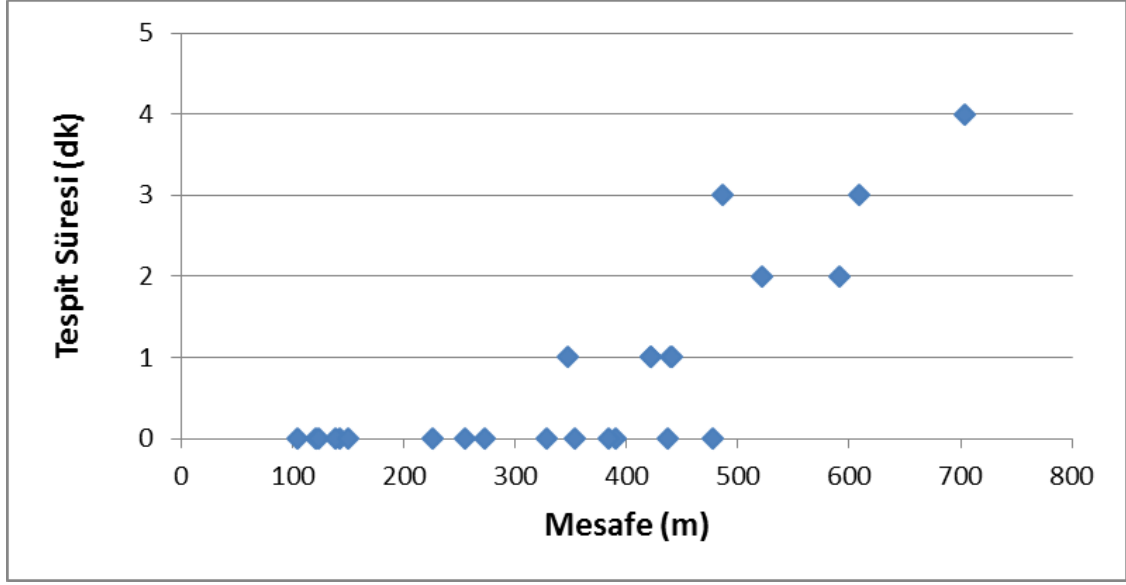
*Şekil 9. Kaliforniya Algoritması için düşük trafik talebinde, olayların tespit zamanı vs olayların yukarıakış sensöründen olan uzaklığı*



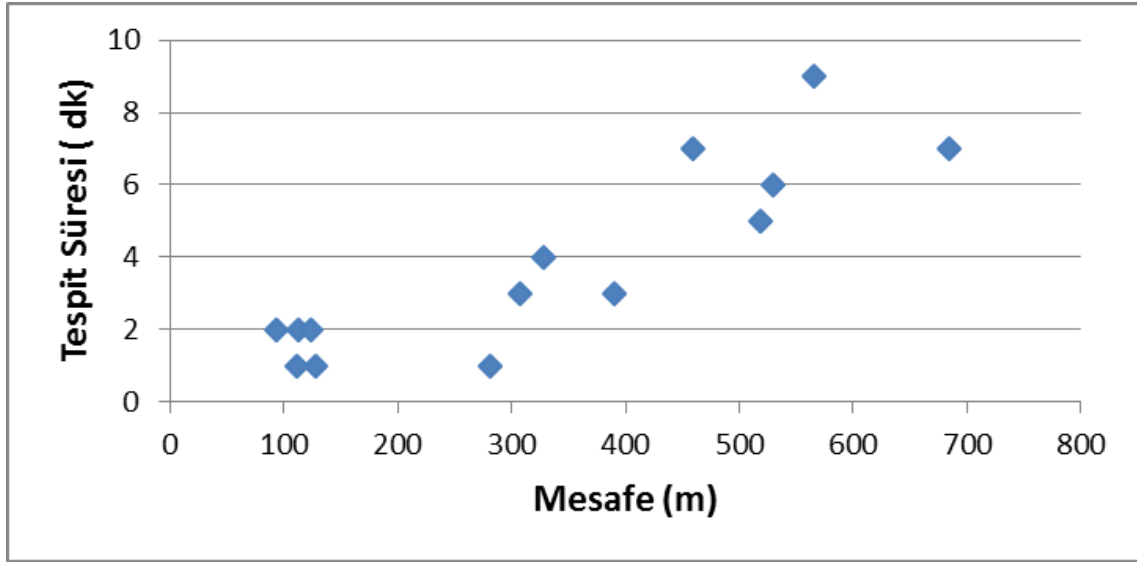
*Şekil 10. Kaliforniya Algoritması için orta trafik talebinde, olayların tespit zamanı vs olayların yukarıdaki sensöründen olan uzaklığı*



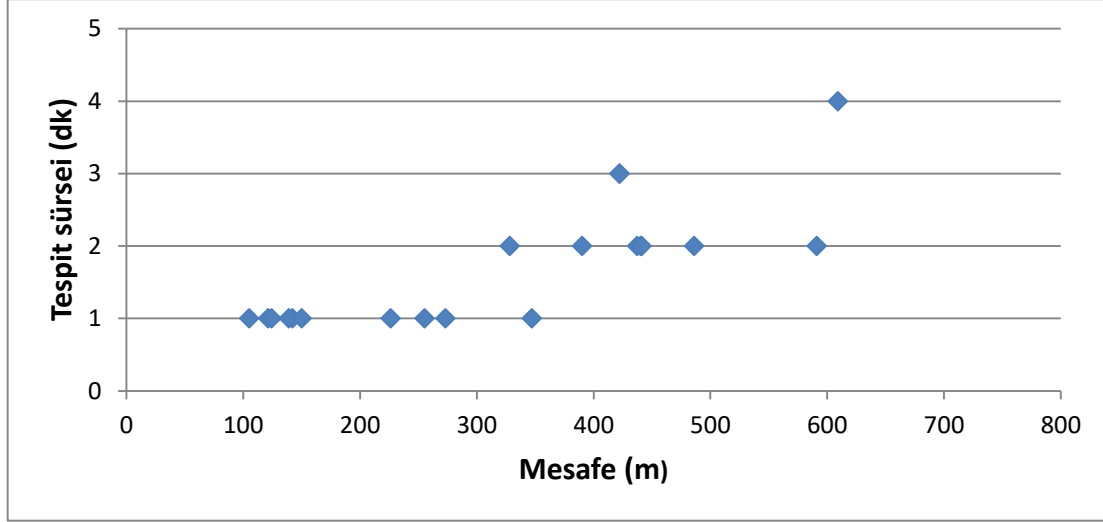
*Şekil 11. Minnesota Algoritması için düşük trafik talebinde, olayların tespit zamanı vs olayların yukarıdaki sensöründen olan uzaklığı.*



*Şekil 12. Minnesota Algoritması için orta trafik talebinde, olayların tespit zamanı vs. olayların yukarıdaki sensöründen olan uzaklığı.*



*Şekil 13. SNS Algoritması için düşük trafik talebinde, olayların tespit zamanı vs. olayların yukarıdaki sensöründen olan uzaklığı.*



*Şekil 14. SNS Algoritması için orta trafik talebinde, olayların tespit zamanı vs olayların yukarıdaki sensöründen olan uzaklığı.*

## 5. Sonuçlar

Kaliforniya, Minesota ve SNS Algoritmaları D100 Karayolunun 2.5 km'lik bölümü PARAMICS mikro simülasyon yazılımı kullanılarak modellenmiştir. PARAMICS modeli ile değerlendirilen üç algoritmanın da İstanbul trafiği için çok uygun olduğu gösterilmiştir. Kaliforniya, Minesota ve SNS Algoritmaları orta trafik talebinde DR değerlerini sırasıyla %25, %31,5 ve %25 yükselmiştir. MTTD değerleri orta trafik talebinde sırasıyla 48.99%, 59.26% ve 48.80% azalmıştır. Minesota Algoritması en düşük MTTD (0.88) değerinde çok iyi performans göstermektedir. Bunun yanında, SNS Algoritması düşük trafik düzeyinde sadece 1 yanlış alarm verilmiştir. Tüm test edilmiş algoritmalar, dedektörler arası eşit ve 800m'den az olduğunda Trafik Kontrol Merkezinden alınan gerçek zamanlı trafik verileri kullanılarak uygulanabilir.

## TEŞEKKÜR

Bu çalışma **Boğaziçi Üniversitesi BAP Komisyonu 09A401P kodlu** proje kapsamında yapılmış ve desteklenmiştir.

## Kaynaklar

- [1].Tina Collier, Ginger Goodin, *Managed Lanes: A Cross-Cutting Study*, No. FHWA-HOP-05-037, USA, 2004.
- [2].Kaan Ozbay, Pushkin Kachroo, *Incident Management In Intelligent Transportation Systems*, Electrical and Computer Engineering Faculty Publications, USA, 1999.
- [3].Payne, H. J; Helfenbein, E. D; Knobel, H. C, *Development and testing of incident detection algorithms, volume 2: Research methodology and detailed results*, Report No. FHWA-RD- 76-20, USA, 1976.
- [4].Payne, H. J; Tignor, S. C, *Freeway Incident-Detection Algorithms Based On Decision Trees With States*. TRB, National Research Council, 1978, pp. 30-37.
- [5].Mak, Chin Long; Henry SL Fan, Transferability of expressway incident detection algorithms to Singapore and Melbourne. *Journal of Transportation Engineering* 2005, 31, 2, 101-111.
- [6].Mountain Plains Consortium (MPC), *Evaluation of Incident Detection Algorithms* Report No: MPC-01-122, USA, 2011.
- [7].*Paramics Traffic Microsimulation Software Manual Version 6.7.2.*, USA, 2011.
- [8].Chassiakos, A. P; Stephanedes, Y. J, Smoothing algorithms for incident detection. *Transportation Research Record*. 1993, 1360, 50-57.