

Dinamik Bayes Ağ Yapısı ve Genişletilmiş Kalman Filtresi Kullanılarak Gerçek Zamanlı ROS Uygulaması ile Otonom Bir Araçtaki Yanal Dinamiklerdeki Arıza Tespitinin Gerçeklenmesi

Fault Diagnosis of the Vehicle Lateral System Using Bayesian Networks and EKF with Real-Time ROS Applications

Mustafa Fatih Yalçın¹, Tolga Bodrumlu¹, Murat Gozum¹, Eyup Ates¹

¹AVL Research and Engineering
Istanbul, Turkey

fatih.yalcin@avl.com, tolga.bodrumlu@avl.com, murat.gozum@avl.com,
eyup.ates@avl.com

Özetçe

Çeşitli mühendislik uygulamalarında, hesaplamalı maliyetleri düşürmek için deneysel modellerle analitik modelleri değiştirmek amacıyla model tabanlı hata teşhis teknikleri kullanılmaktadır. Bayes Ağı tabanlı yöntemler, sağlamlıkları nedeniyle popülerlik kazanmıştır. Bu makale, bir aracın yanal dinamik sistemine yönelik hataları tespit etmek için Bayes Ağı ve Genişletilmiş Kalman Filtresi'ni birleştiren model tabanlı bir hata teşhis yaklaşımını tanıtmaktadır. Savrulma oranı, tekerlek kayma oranı ve direksiyon açısı artık değerleri, sensör verilerini analitik modellerden elde edilen verilerle karşılaştırarak hesaplanır. Aracın hızı, GPS ve ivmeölçer verileri kullanılarak Genişletilmiş Kalman Filtresi (EKF) ile tahmin edilir ve teker hız sensöründeki potansiyel hatalar tespit edilir. Eğer hatalı bir teker hızı ölçümü varsa, bu durum tespit edilir ve sensör füzyonundan elde edilen hız değeri dinamik modele aktarılır. Önerilen yöntem önce Matlab/Simulink ortamında modellenir ve test edilir. Daha sonra bu algoritma C++ dilinde uygulanır ve ROS (Robot İşletim Sistemi) ortamına gömülür, iletişim yapılandırması sağlanır. Ayrıca, gerçek zamanlı test sırasında hataları görsel olarak izlemek için bir İnsan-Makine Arayüzü (HMI) ekranı geliştirilir. Gerçek zamanlı testlerde, tasarlanan algoritmanın yüksek doğrulukla hata tespit edebildiği gözlemlenir.

Abstract

In various engineering applications, model-based fault diagnostic techniques are employed to reduce computational costs by replacing analytical models with experimental models. Bayesian Network-based methods have gained popularity for their robustness. This paper introduces a model-based fault diagnostic approach that combines Bayesian Network and Extended Kalman Filter to identify faults in the lateral dynamic system of a vehicle. The values of yaw rate, wheel slip rate, and steering angle residual are calculated by comparing sensor data with data obtained from analytical models. The vehicle's speed is estimated using GPS and accelerometer data through an Extended Kalman Filter (EKF) to detect potential faults in the wheel speed sensor. If there is a faulty wheel speed measurement, this situation is detected and the speed value

obtained from the sensor fusion is used in the dynamic model. The proposed approach is initially modeled and tested in the Matlab/Simulink environment. Then this algorithm is implemented in C++ and embedded into the ROS (Robot Operating System) environment, with communication structure. Additionally, a Human-Machine Interface (HMI) screen is developed to visually monitor faults during real-time testing of the algorithm. During the real-time tests, it is observed that the designed algorithm can detect the fault with a high accuracy.

1. Giriş

Yenilikçi arıza tespit ve teşhis yöntemleri [1-4] sayesinde teknik süreçlerin güvenliği ve güvenilirliği gelişmiştir. Model tabanlı arıza tespiti, karmaşık sistemlerdeki arızaların etkin bir şekilde tespit edilmesini sağlayan etkili bir yöntemdir [5]. Bu yöntem, sistemlerin matematiksel modellerini kullanarak normal işleyişten sapmaları tespit eder ve arızaların nedenlerini belirlemeye yardımcı olur. Bu tekniklerde, analitik modellerden hesaplanan değer ile ölçülen değer analitik yedeklilik kullanılarak karşılaştırılır [6]. Model tabanlı arıza tespitleri, analitik yedeklilikleri sayesinde güvenilir sonuçlar sağlarken, ek maliyet ve ağırlık yükü olmadığı için maliyet açısından avantajlıdır. Ancak doğru modelleme ve güncellemelerin önemi unutulmamalıdır. Sonuç olarak, bu yöntem, teknik süreçlerin güvenliğini ve güvenilirliğini artırmada önemli bir rol oynamaktadır. Arıza ağaçları, arızaların teşhis edilmesinde yaygın olarak kullanılan bir başka yöntemdir. Bu yöntemde, kullanıcıya olası arıza belirtilerini ifade eden bir dizi soru sorulur ve kullanıcının bu sorulara verdiği cevaplara göre arızanın nedeni belirlenmeye çalışılır [7-8]. Arıza ağaçları, pratik ve kullanıcı dostu bir yaklaşım olsa da, bazı önemli dezavantajları vardır. Örneğin, belirli bir arıza nedeni hakkında yargının belirsiz olduğu durumlar model tarafından ele alınmaz. Ayrıca, ağacın sabit yapısı uzmanlık veya önceki bilgilerin teşhis sürecine entegre edilmesini engeller ve birden fazla semptomlu arızaları tespit etmekte zorlanabilir [9]. Sinyal tabanlı arıza tespit sistemleri, karmaşık sistemlerde arızaların tespit edilmesinde kullanılan etkili bir yöntemdir [10]. Bu yöntem, sistemin ölçülen çıkış sinyallerini analiz ederek

arızaları tespit eder. Sistemdeki arızalar genellikle ölçülen sinyallerin özelliklerindeki değişikliklerle ilişkilendirilir ve bu değişiklikler aracılığıyla tespit edilir. Sinyal tabanlı arıza tespiti, sistemin matematiksel modelini kullanmaktan bağımsızdır, bu nedenle sistem modellemesinin karmaşıklığına ve belirsizliğe ihtiyaç duymaz [11]. [12]'de yazarlar, bir asenkron motorda arıza teşhisi gerçekleştirmek için model tabanlı ve sinyal tabanlı yaklaşımlar kullanmışlar ve kullanılan modellerin karmaşıklığı nedeniyle model tabanlı yaklaşımın uygulanmasının sinyal tabanlı yaklaşıma göre daha zor olmasına rağmen daha iyi performans gösterdiği sonucuna varmışlardır.

Bu bildiriye, sabit olmayan boylamsal hızı sahip otonom bir aracın yanal dinamiği için model tabanlı bir arıza tespit algoritması geliştirilmiştir. Geliştirilen arıza tespit algoritması hem Bayes Ağı hem de Genişletilmiş Kalman Filtresi (GKF) içermektedir. Algoritmada kullanılan dinamik model, araç hız verilerinin entegrasyonu ile çalıştığından, hız verilerinin tek bir hatalı ölçümü bile tüm sistemin dinamik yapısını bozabilmektedir. Bu sorun GKF yapısı kullanılarak çözülmüştür. Arıza tespit algoritmasında altı artık değer hesaplanmaktadır. İlk üç artık değer, jiroskop ile ölçülen savrulma hızı ile sırasıyla ön tekerleklerden, arka tekerleklerden ve dinamik bisiklet modelinden elde edilen savrulma hızı değerleri karşılaştırılarak hesaplanmıştır. Sağ ve sol tekerleklerin sensör ölçümlerinden elde edilen kayma oranı ile lastik modelinden elde edilen kayma oranı karşılaştırılarak iki ek artık değer daha hesaplanmıştır. En son artık değer, direksiyon açısı sensöründen elde edilen değer ile direksiyon açısı modelinden hesaplanan direksiyon açısı arasındaki fark alınarak hesaplanır. Eğer hatalı bir tekerlek hız ölçümü varsa bu durum tespit edilir ve dinamik modelde sensör füzyonundan elde edilen hız değeri kullanılır. Hesaplanan tüm artıkların eşik değerleri, otonom test aracından toplanan veri kümeleri kullanılarak belirlenmiştir. Artıkların eşik değerini aşmış olduğuna bağlı olarak, hangi sensör veya eyleyici de arıza olduğuna karar vermek için Bayes Ağının katsayıları dinamik olarak güncellenir. Sonuçlar bize Bayes Ağı ve GKF kullanılarak sağlam ve doğru arıza tespitinin sağlandığını göstermektedir. Ayrıca, bu arıza tespit algoritması HMI arayüzü ile görselleştirme yapılarak kullanıcı dostu bir sistem olarak tasarlanmıştır.

2. Arıza Tespit Yapısı

2.1 Artık Değerlerin Elde Edildiği Sensörler

Bu bildiriye, savrulma oranı, kayma oranı ve direksiyon açısı için artık değerler elde edilmiştir. Tablo 1'de hem kullanılan analitik modellerin içerdiği değerlerin hem de elde edilen ölçümlerin yapıldığı sensörlerin sembolleri verilmiştir.

Tablo 1. Sensör ile ölçülen parametrelerin sembolleri ve açıklamaları

Sembol	Açıklama
V_{FL}	Ön Sol Tekerlek Hızı
V_{FR}	Ön Sağ Tekerlek Hızı
V_{RL}	Arka Sol Tekerlek Hızı
V_{RR}	Arka Sağ Tekerlek Hızı
δ	Ön tekerlek açısı
ψ	Savrulma oranı

2.2 Kullanılan Modeller

Bu çalışmada hata ayıklamayı gerçekleştirmeyi sağlayan altı artık değer altı farklı modelden elde edilmiştir. Bu modellerin detayları [13] makalesinde verilmiştir. Bu modellerden üç tanesi savrulma oranını hesaplamak için kullanılmıştır. İki model tekerlek kaymasını hesaplamada, geriye kalan son model de direksiyon açısının hesaplanmasında kullanılmıştır. Savrulma oranının hesaplanmasında ilk olarak doğrusal bisiklet modelinden faydalanılmıştır. Doğrusal bisiklet modeli durum-zay denklemi Eş.1 ile gösterilmektedir [14]:

$$\begin{bmatrix} \dot{y} \\ \dot{y} \\ \dot{\psi} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{af} + 2C_{ar}}{mV_x} & 0 & -V_x - \frac{2C_{af}l_f - 2C_{ar}l_r}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2l_r C_{af} - 2l_r C_{ar}}{I_z V_x} & 0 & -\frac{2l_f^2 C_{af} + 2l_r^2 C_{ar}}{I_z V_x} \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \\ \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2C_{af}}{m} \\ 0 \\ \frac{2l_r C_{af}}{I_z} \end{bmatrix} \delta \quad (1)$$

Bisiklet modelinin dışında ön ve arka tekerlek hız modellerine bağlı savrulma oranı da hesaplanmıştır. Bu modeller Eş. 2 ve Eş. 3'te gösterilmiştir [15]:

$$\dot{\psi} = (V_{FR} - V_{FL})/b_f \quad (2)$$

$$\dot{\psi} = (V_{RR} - V_{RL})/b_r \quad (3)$$

Tekerlek kayma oranının hesaplanmasında ön ve arka tekerleklerde çekiş olmasına bağlı ilişki kullanılmıştır [16]. Bu ilişki Eş. 4'te gösterilmiştir:

$$S = \frac{\omega_{undriven} - \omega_{driven}}{\omega_{undriven}} \quad (4)$$

Lineer durum varsayımıyla, tekerlek kayma oranı Eş. 5 kullanılarak da hesaplanabilir [17]:

$$F_x = \frac{1}{2} m a_x = C_x S \quad (5)$$

Direksiyon açısının hesaplanmasında ise Eş. 6 kullanılmıştır:

$$\delta = \frac{l \cdot i_{st}}{V_x} \dot{\psi} \quad (6)$$

2.3 Arıza Ayıklama Planı

Savrulma oranına ait üç tane artık değeri, modellerden hesaplanan değerler ile sensörden okunan savrulma oranlarının birbirinden çıkarılmasıyla elde edilmiştir:

$$\begin{aligned} R_1 &= \hat{\psi}_{model_1} - \psi_{sensor} \\ R_2 &= \hat{\psi}_{model_2} - \psi_{sensor} \\ R_3 &= \hat{\psi}_{model_3} - \psi_{sensor} \end{aligned} \quad (7)$$

Burada model₁, model₂, ve model₃ sırasıyla bisiklet, ön tekerlek ve arka tekerlek hız modellerini temsil etmektedir. Aracın her iki tarafında çekiş olan tekerleklerin kuvvet ilişkisinden elde edilen kayma oranlarının değerlerini (Eş. 5), sol ve sağ tekerlek hızlarından elde edilen kayma oranlarının değerlerinden (Eş.4) çıkardığımızda iki tane daha artık değer elde edilir:

$$R_4 = S_{force\ relation} - S_{right\ wheel\ speeds} \quad (8)$$

$$R_5 = S_{force\ relation} - S_{left\ wheel\ speeds}$$

Direksiyon açısının hesaplandığı modelin (Eş. 6), direksiyon açısının elde edildiği sensör ölçümüyle olan farkından altıncı ve artık değer elde edilmiştir:

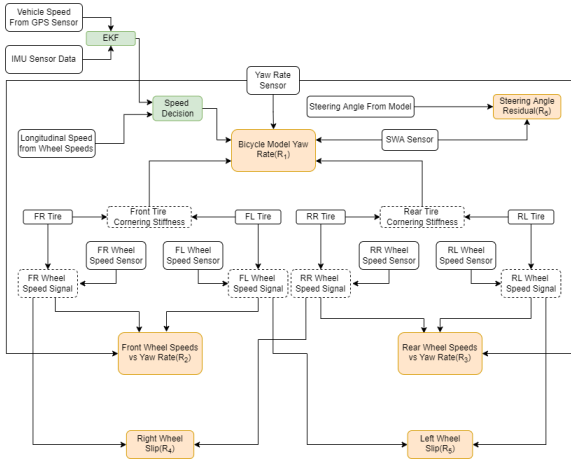
$$R_6 = \hat{\delta}_{model} - \delta_{sensor} \quad (9)$$

Bunlardan ayrı olarak altı tane sensörden kaynaklı ve dört tane fiziksel olmak üzere toplamda on tane arıza tanımlanmıştır. Bu arızalar ve açıklamaları Tablo 2’de verilmiştir.

Tablo 2. Arızalar ve açıklamaları (İlk dört tanesi fiziksel ve son altı tanesi sensörden kaynaklı arıza).

Hata	Açıklama
F_1	Ön sağ lastik arızası
F_2	Ön sol lastik arızası
F_3	Arka sağ lastik arızası
F_4	Arka sol lastik arızası
F_5	Ön sağ tekerlek hızı sensör arızası
F_6	Ön sol tekerlek hızı sensör arızası
F_7	Arka sağ tekerlek hızı sensör arızası
F_8	Arka sol tekerlek hızı sensör arızası
F_9	Savrulma oranı sensör arızası
F_{10}	Ön tekerlek açısı sensör arızası

Tabloda gösterilen arızaları, hesap edilen artık değerlerine bağlayan Bayes ağ yapısı Şekil 1’de gösterilmiştir.



Şekil 1. Arıza Tespit Algoritma Şeması

2.4 Dinamik Bayes Ağı

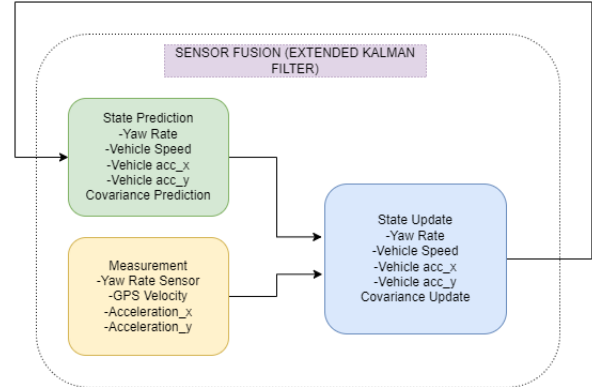
Bu çalışmada, arıza olasılıkları Bayes ağı kullanılarak hesaplanmıştır. Kullanılan yapıya göre, herhangi bir artık değer aktif olduğunda, ona bağlı arızaların olasılıkları dinamik olarak güncellenmiştir. Şekil 1’de de gösterildiği üzere her bir artık değeri etkileyen arızalar bulunmaktadır. Bu oluşturulan dinamik Bayes ağının mantığı ve detaylı açıklaması [13]’te verilmiştir. Tablo 3’te belirli artık değerler için hangi arızaların aktif olduğu gösterilmiştir

Tablo 3. Arızalar ve Onların Bağlı Olduğu Artık Değerleri

R_1	R_2	R_3	R_4	R_5	R_6	Arızalar
1	0	0	0	0	1	F_{10}
1	0	1	0	1	0	F_4
1	1	0	1	0	0	F_1
1	1	0	0	1	0	F_2
1	0	1	1	0	0	F_3
1	0	1	0	1	1	F_8, F_{10}
1	1	0	0	1	1	F_6, F_{10}
1	1	0	1	0	1	F_5, F_{10}
1	0	1	1	0	1	F_7, F_{10}
1	1	1	0	0	1	F_9, F_{10}

2.5 Genişletilmiş Kalman Filtresi

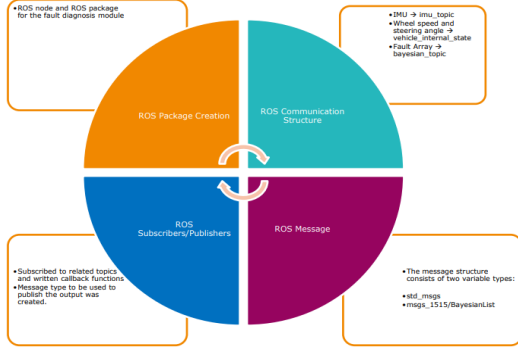
Bisiklet modelinin girdilerinden bir tanesi tekerlek sensörlerinden gelen hız verisi ile hesaplanan araç hızıdır. Bundan ötürü, herhangi bir tekerlekten gelecek olan arızalı bir hız ölçümü, bisiklet modelinin sapma oranı açısını yanlış hesaplamasına sebep olacaktır. Bu durumda, arıza tespit algoritması, esasen hız verisinden kaynaklı arıza olmasına rağmen, arızanın sapma oranında olduğunu tespit etmektedir. Bu problemi çözebilmek için, oluşturmuş olduğumuz arıza tespit algoritması, GPS sensöründen alınan hız verileri ile IMU sensöründen alınan ivmeölçer verilerini Genişletilmiş Kalman Filtresi (GKF) yardımıyla birleştirerek aracın hız verisini elde etmektedir. Bu noktada, bisiklet modeli tarafından elde edilen sapma oranının yanlış hesaplanmasını engellemek için tekerlek hız sensörlerinden gelen hız değeri yerine bu birleştirilmiş hız verisinin kullanılması gerekmektedir.



Şekil 2. GKF Sistem Şeması

2.6 ROS Yapısı

Bölüm 2.4’te açıklanan Bayes Ağ yapısı otonom bir test aracına hem simülasyon ortamında hem de gerçek ortamda uygulanmış ve test edilmiştir. Bunu yaparken ROS (Robotik İşletim Sistemi)’nin Melodic sürümü Ubuntu v18.04 üzerinde kullanılmıştır.

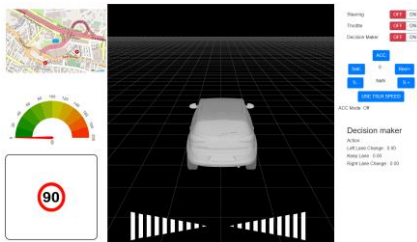


Şekil 3. Arıza Tespitinde Kullanılan ROS Yapısı

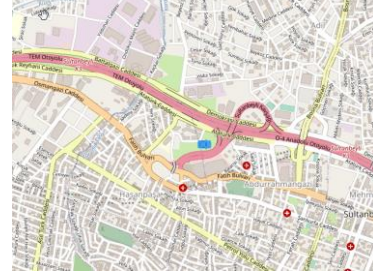
Her bir arızanın olasılıksal çıktısını elde etmek amacıyla Arıza Tespiti düğümünü içeren bir ROS paketi oluşturulmuştur. Bu doğrultuda düğümün her bir döngüsünün sonunda artık değerleri hesaplayan fonksiyonlar yazılmıştır. Bu artık değerlerin hesaplanmasında kullanılması gerekli sensör okumalarına ilişkili sensörün konusuna (topic) abone olunarak ulaşılmıştır. Artık değerlerin hesaplanmasının akabinde bu değerler Bayes Ağını oluşturan ve sonrasında arıza olasılıklarını hesaplayan başka bir fonksiyona girdi olarak iletilmiştir. Son olarak bu olasılıklar, her birini görünür kılacak şekilde oluşturulan bir mesaj tipi yaratılarak Bayes konusuna yayınlanmıştır. Uygulanan ROS yapısı Şekil 3'te de görsel olarak özetlenmiştir. Bir sonraki bölümde simülasyon sisteminde yapılan çalışmalar aktarılacaktır.

2.7 HMI Yapısı

İnsan-Makine Arayüzü (HMI), insan ve araç arasındaki temel etkileşimi sağlayan, iletişimi CAN, ROS vb. iletişim yöntem ve protokolleri üzerinden gerçekleştiren yazılım ya da donanım olarak tanımlanabilir. Modern HMI'lar, sezgisel kontrol ve ekran sistemleri sağlamak üzere tasarlanmıştır ve eğlence, navigasyon ve otonom sürüş destek sistemleri gibi çeşitli fonksiyonları ihtiva eder. Ayrıca, kullanıcı HMI aracılığıyla temel bazı sensör çıktılarını gözlemleyebilir. Geliştirdiğimiz HMI'nın kullanım sıklığı ve birbiriyle ilintisine göre gruplandırılmış 4 ana sayfası bulunmaktadır. İlk sayfa, şerit ve yol bilgilerini, çevresel engelleri, TSLR (Trafik Işığı ve Levha Tanıma) bilgilerini, karar vericinin kararlarını ve olasılıklarını, küçük bir navigasyon haritasını ve ACC (Adaptif Hız Sabitleyici) kontrollerini içeren ana HMI sayfasını içerir (Şekil 4). İkinci sayfa; istenen ve gerçek hız ile gerçek ve istenen direksiyon açılarını gösteren gösterge sayfasıdır. Üçüncü sayfa, ana harita ve navigasyon sayfasıdır (Şekil 5) ve son olarak hata teşhis sayfası bulunur. Proje, reactJS teknolojisi kullanılarak geliştirilmiştir. HMI ve araç arasındaki tüm veri iletişimi ROS teknolojisi üzerinden sağlanmıştır.



Şekil 4. Ana HMI sayfası



Şekil 5. Harita ve navigasyon sayfası

Bir sonraki bölümde gerçekleştirilen test sonuçları ve detayları aktarılacaktır.

3. Test Senaryoları

ROS içerisine entegre edilen arıza tespit algoritmasının testleri gerçek araç verisi kullanılarak gerçekleştirilmiştir. Gerçek araç üzerinde toplanan verilere test sırasında, o an istenilen sensörde arıza oluşturabilecek şekilde değerler eklenmiştir. Burada seçilen test senaryoları sırasıyla sağ ve sol ön tekerlek hız sensörleridir. Burada hız sensörlerine arıza verilmesinin temel nedenlerinden birisi de hız sensörlerinde bir arıza olduğunda Bayes Ağının bu durumu tespit edebilmesi ve akabinde dinamik bisiklet modelinde tekerlek hız sensöründen gelen hız değeri yerine GKF ile elde edilen hız değerinin kullanılmaya başlanmasıdır.

Arıza tespiti gerçekleştirebilmek için her bir artık değer için eşik değeri belirlenmiştir. Bu eşik değeri belirlenmesi sırasında yine gerçek araç test verileri kullanılmıştır. On adet araç verisi incelenerek (arıza olmadığı kabul edilen) her bir artık değer için eşik değerler oluşturulmuş ve Tablo 4 içerisinde bu değerler gösterilmiştir. Hesaplanan artık değerlerin bu eşik değerleri aştığı durumlarda o artık değer 1, aşmadığı durumlarda ise 0 olarak ataması gerçekleştirilecektir.

Tablo 4. Artık Değer Eşik Değeri

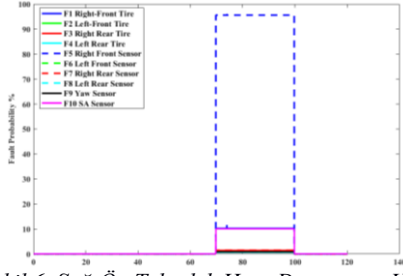
Artık Değer	Eşik Değeri
R ₁ , R ₂ , R ₃	0.2 rad/s
R ₄ , R ₅	0.02
R ₆	0.4 rad

3.1. Test Senaryoları

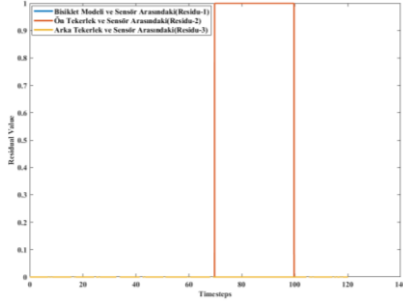
Hazırlanan ilk test senaryosu sağ ön tekerlek hız sensörü için, ikinci test senaryosu ise sol ön tekerlek hız sensörü için gerçekleştirilmiştir.

a) Sağ Ön Tekerlek Hız Sensörü

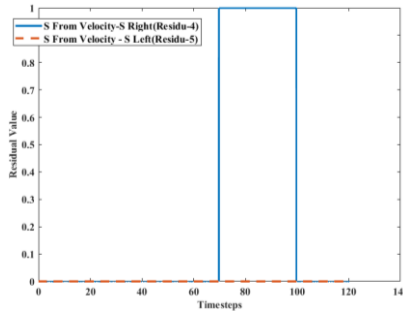
Sağ ön tekerlek hız sensörü 70 ile 100. saniye arasında arızalanmıştır. Sağ ön tekerlek sensöründeki bir arızadan dolayı burada bir arıza olduğu ve bunun yerine GKF'den gelen hızın kullanıldığı tespit edilmiştir. Bisiklet modeli içerisinde tekerlek sensöründen gelen hız yerine, GKF'den gelen hız kullanıldığı için R₁'in artık değeri eşik değerini aşmamış ve R₁'de herhangi bir arıza oluşmamıştır. Bu durum hata analizini doğru yapılmasını sağlamış olup, sistemin sağ ön tekerlek sensör hatasına düşmesini sağlamıştır. İlgili hata durumu arayüz ekranında da gösterilmiştir.



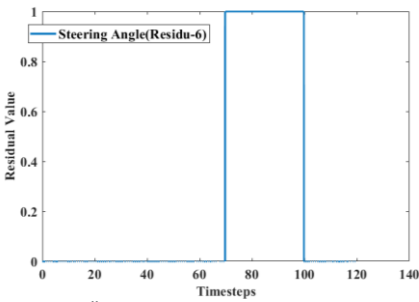
Şekil 6. Sağ Ön Tekerlek Hata Durumu ve Hata Olasılıkları



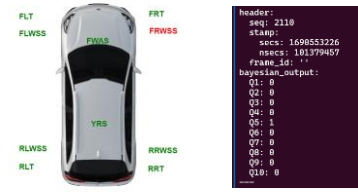
Şekil 7. Sağ Ön Tekerlek Hata Durumu ve R1-R2-R3 Artık Değerleri



Şekil 8. Sağ Ön Tekerlek Hata Durumu ve R4-R5 Artık Değerleri



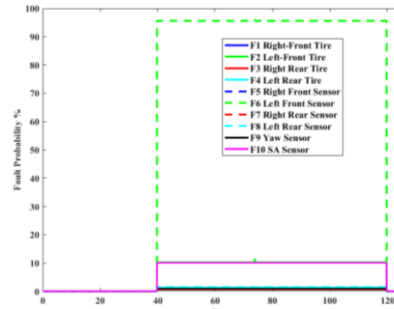
Şekil 9. Sağ Ön Tekerlek Hata Durumu ve R6 Artık Değeri



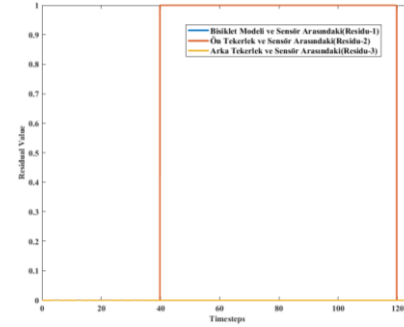
Şekil 10. Sağ Ön Tekerlek Hata Durumu ve HMI Görüntüsü

b) Sol Ön Tekerlek Hız Sensörü

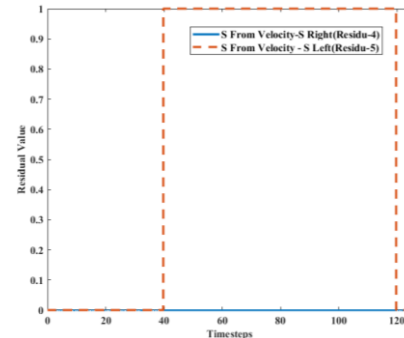
Sol ön tekerlek hız sensörü 40 ile 120. saniye arasında arızalanmıştır. Sağ ön tekerlek sensöründeki bir arızadan dolayı burada bir arıza olduğu ve bunun yerine GKF'den gelen hızın kullanıldığı tespit edilmiştir. Bisiklet modeli içerisinde tekerlek sensöründen gelen hız yerine, GKF'den gelen hız kullanıldığı için R_1 'in artık değeri eşik değerini aşmamış ve R_1 'de herhangi bir arıza oluşmamıştır. Bu durum hata analizini doğru yapılmasını sağlamış olup, sistemin sol ön tekerlek sensör hatasına düşmesini sağlamıştır. İlgili hata durumu arayüz ekranında da gösterilmiştir.



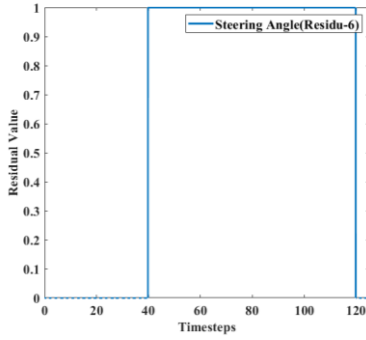
Şekil 11. Sol Ön Tekerlek Hata Durumu ve Hata Olasılıkları



Şekil 12. Sol Ön Tekerlek Hata Durumu ve R1-R2-R3 Artık Değerleri



Şekil 13. Sol Ön Tekerlek Hata Durumu ve R4-R5 Artık Değerleri



Şekil 14. Sol Ön Tekerlek Hata Durumu ve R6 Artık Değeri



Şekil 15. Sol Ön Tekerlek Hata Durumu ve HMI Görüntüsü

4. Sonuçlar

Bu çalışmada, dinamik olarak koşullu olasılık dağılımları güncellenen model tabanlı arıza tespit eden bir yöntem anlatılmıştır. Önerilen bu yöntem hem Bayes ağ yapısını hem de GKF'yi kullanmaktadır. Bu noktada, aracın hız verisi GKF yardımıyla, GPS sensörü ve IMU sensöründen elde edilen verilerin birleştirilmesiyle elde edilmektedir. Herhangi bir tekerlek sensöründen arızalı bir ölçüm geliyorsa, GKF yardımıyla elde edilen hız verisi tekerlek hız verisinin yerine kullanılarak bisiklet modeli tarafından hesaplanan savrulma oranının hatalı bir şekilde hesaplanmasının önüne geçecektir. Jiroskop sensörü tarafından ölçülen savrulma oranı, üç tane analitik model (ön tekerlek, arka tekerlek ve bisiklet dinamik modeli) tarafından hesaplanan savrulma oranlarıyla karşılaştırılmıştır. Savrulma oranından başka, sol ve sağ tekerleklerin ölçümleri kullanılarak elde edilen kayma oranı, tekerlek modeli tarafından hem ön hem de arka tekerlek için hesaplanan kayma oranlarıyla karşılaştırılmıştır. Buna ek olarak, direksiyon açısı sensöründen elde edilen değer ile modelden hesaplanan direksiyon açısı karşılaştırılmıştır. Bu sensör ve model değerleri karşılaştırılması sonucu altı tane artık değer elde edilmiştir. Gerçek araç verilerini kullanarak, ROS içine entegre ettiğimiz dinamik Bayes ağını kullanarak, sol ve sağ ön tekerlek hız sensörlerindeki hatalar başarıyla tespit edilmiştir. Ayrıca, hatalar tekerlek hız sensörlerinde olduğu için sisteme entegre etmiş olduğumuz GKF'nin yardımıyla aracın savrulma oranı tüm sürüş sırasında doğru olarak tahmin edilmiştir. Sonuç olarak, gerçek zamanlı testlerde hem Bayes ağı hem de GKF'yi kullanarak oluşturmuş olduğumuz algoritma, aracın yanal dinamikleriyle bağlantılı hatalar başarılı bir şekilde tespit edilmiş ve arayüzde başarılı bir şekilde gösterilmiştir.

Kaynakça

- [1] Ko, C., & Fox, D. (2009). GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. The International Journal of Robotics Research, 28(12), 1524-1547.
- [2] Wang, Z., Gao, Z., & Ding, S. X. (2012). A survey of model-based fault detection and diagnosis methods. Acta Automatica Sinica, 38(6), 823-839.
- [3] Chen, J., & Patton, R. J. (1999). Robust model-based fault diagnosis for dynamic systems. Kluwer Academic Publishers.
- [4] Raghuraman, S., & Mahadevan, S. (2010). Fault detection and diagnosis using dynamic Bayesian networks and system identification. Journal of Process Control, 20(5), 604-616.
- [5] S.X. Ding, "Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools", Berlin, Germany: Springer-Verlag, 2008.
- [6] J. Marzat, H.P.Lahanier, F. Damongeot, and E. Walter "Model-based fault diagnosis for aerospace systems: a survey" Proc. the Institution of Mechanical Engineers Part G Journal of Aerospace Engineering, vol. 226, pp.1329-60, 2011.
- [7] Ji, L., Zhang, Y., Yan, R., & Song, Y. (2019). Fault diagnosis of power transformers based on fault tree and fault tree-analytic hierarchy process. Electric Power Systems Research, 174, 105822.
- [8] Ji, L., Zhang, Y., Yan, R., & Song, Y. (2019). Fault diagnosis of power transformers based on fault tree and fault tree-analytic hierarchy process. Electric Power Systems Research, 174, 105822.
- [9] Li, R., Song, Y., & Song, H. (2016). Fault tree-based fault diagnosis method for nuclear power plant passive safety systems. Annals of Nuclear Energy, 87, 260-267.
- [10] Huang, B., Zhu, Q., Zhang, D., & Li, X. (2019). A survey of signal processing techniques for rotating machinery fault diagnosis. Measurement, 136, 597-619.
- [11] Xu, Z., Liang, M., Chen, X., & Sun, Y. (2017). A survey of data-driven approaches for condition monitoring and fault diagnosis of electrical machines. IEEE Transactions on Industrial Electronics, 65(5), 3990-4001.
- [12] P. P. Harihara, Kyusung Kim and A. G. Parlos, "Signal-based versus model-based fault diagnosis-a trade-off in complexity and performance," 4th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, 2003. SDEMPED 2003., 2003, pp. 277-282.
- [13] T.Bodrumlu, M.Gozum, and Batıkan Kavak, "Dinamik Bayes Ağ Yapısı ve Genişletilmiş Kalman Filtresi Yardımıyla Otonom Bir Araçtaki Yanal Dinamiklerdeki Arıza Tespitinin Gerçeklenmesi", Fırat Üniversitesi Uzay ve Savunma Teknolojileri Dergisi, 1(1), 362-369, 2022.
- [14] M. Ciboğlu, "Hybrid Controller Approach for an Autonomous Ground Vehicle Path Tracking Problem", M.Sc. Thesis, Dept. Cont. Autom. Eng., Istanbul Technical University, Istanbul, Turkey, 2016.
- [15] D.Fischer, M. Börner, J. Schmitt and R. Isermann, "Fault detection for lateral and vertical vehicle dynamics" Control Engineering Practice, vol. 15, pp.315-324, 2007.
- [16] M. L. Schwall, and J.C. Gerdes, "Multi-modal Diagnostics for Vehicle Fault Detection," in Proc. IMECE2001. ASME International Mechanical Engineering Congress and Exposition, New York, NY, United States, 2001.
- [17] Miller, S. et al, "Calculating Longitudinal Wheel Slip and Tire Parameters Using GPS Velocity", Proceedings of the American Controls Conference, 2001, pp 1800-1805.