Kooperatif Sinir Ağları ile Akıllı Direksiyon Kontrolcüsü Geliştirme
Developing Intelligent Steering Controller with Cooperative Neural Networks

Erkin Dinçmen

Kontrol ve Otomasyon Mühendisliği Bölümü
Yıldız Teknik Üniversitesi, Davutpaşa, İstanbul
erkin.dincmen@yildiz.edu.tr

Özetçe

Bu çalışmada iki yapay sinir ağından oluşan bir direksiyon kontrol sistemi tanıtılmıştır. İki yapay sinir ağlı yapı sayesinde kontrol edilen sistemde var olan veya sonradan oluşabilecek bozuculara, düzgünsüzlüklere, sinyal gecikmelerine dayanıklı bir kontrol sistemi elde edilmiştir. Yapay sinir ağlarından birincisi sistemin dinamiğini öğrenmekte, ikincisi ise birinci ağın yardımı ile kendini sürekli eğiterek kontrol girişlerini üretmektedir. Bu şekilde kendini devamlı yenileyen akıllı bir kontrol sistemi sunulmuştur. Sistemin performansı benzetim çalışmaları ile test edilmiştir. Ayrıca kontrol sisteminin Lyapunov kararlılık analizi bildirinin eki kısmında verilmiştir.

Abstract

In this study, a steering control system consisting of two artificial neural networks is introduced. The two-neural network structure provides a control system robust to existing or future disturbances, non-linearities, and signal delays within the controlled system. The first neural network learns the dynamics of the system, while the second, using the first network, continuously trains itself to generate control inputs. Thus, a continuously self-adapting intelligent control system is presented. The system's performance is tested through simulations. A Lyapunov stability analysis of the control system is also presented in the appendix.

1. Giriş

Otonom araç teknolojisi sayesinde insan hatalarından bağımsız, daha güvenli ve daha verimli bir ulaşım ekosistemi oluşturulması amaçlanmaktadır. Bu teknolojide tekil araçların otonom olarak hareket etmesi söz konusu olduğu gibi ayrıca otonom konvoylarda belirli sayıda araç grubunun koordineli bir şekilde ilerlemesi, böylece trafikte tıkanmaların engellenerek gecikmelerin minimize edilmesi amaçlanmaktadır. Bu teknolojide aracı ve/veya araçları yönlendiren direksiyon kontrolcülerinin performansı, gerek bireysel otonom araçların kararlılığı gerekse konvoy kararlılığı için hayati önemdedir. Örneğin direksiyon sisteminde zamanla oluşabilecek davranış farklılıkları, bozucular, düzensizlikler, kontrol sinyallerinde ve sensör sinyallerinde oluşabilecek gecikmeler kontrolcünün performansını etkileyebilir ve bunun sonucunda otonom araçların trafikte tehlikeli davranışlar göstermelerine sebep olabilir.

Literatürde bu amaçla çeşitli direksiyon kontrol algoritmaları geliştirilmektedir: [1]'de otonom bir araçta parametrik belirsizlikler ve değişken araç hızını göz önüne alan model öngörülü direksiyon kontrolcüsü önerilmiştir. [2]'de modelleme belirsizliklerinin ve bozucuların dikkate alındığı öğrenme tabanlı bir direksiyon kontrolcüsü önerilmiştir. [3]'te direksiyon mekanizmasında boşluk bulunan otonom bir araç için uyarlamalı bir direksiyon kontrolcüsü önerilmiştir. [4]'te otonom aracın direksiyon sisteminde var olan boşluk ile başa çıkmak için değişken kazançlı kayma kipli kontrolör önerilmiştir. [5]'te ağ gecikmesinin neden olduğu direksiyon açısı salınımları ile başa çıkmak için model öngörülü otonom araç direksiyon kontrolcüsü geliştirilmiştir. [6]'da ağ gecikmesine sahip otonom araçların direksiyon kontrolü için olay tetiklemeli hataya dayanıklı kontrol algoritması önerilmiştir. [7]'de önerilen kontrol algoritması direksiyon sinyali gecikmesini göz önüne almaktadır. Bu amaçla bir model öngörülü kontrol algoritması kullanılmıştır. [8]'de direksiyon eyleyicisinde gecikme olan otonom araçların direksiyon kontrolü için doğrusal parametre değişken H_∞ kontrol algoritması önerilmiştir. [9]'da direksiyon saturasyonu ve yol eğriliği kısıtları altında hareket kontrolü sağlayan kayma kipli kontrol algoritması sunulmuştur.

Bu bildiride iki yapay sinir ağından oluşan bir direksiyon kontrol sistemi sunulmuş ve direksiyon eyleyicisinde sinyal gecikmesi olan bir araç modeline uygulanmıştır. [10]'da sunulan çalışma genişletilerek daha gerçekçi bir yol profili takibi için ve direksiyon sisteminde daha büyük sinyal gecikmesi olması durumu için analiz yapılmıştır. Kontrol algoritmasında ilk olarak model ağı ile aracın dinamikleri öğrenilmektedir. Daha sonra kontrolör ağı aracı kontrol etmeye başlamakta, araç kontrol edilirken aynı zamanda kontrolör ağı model ağı ile işbirliği yaparak kendini devamlı eğitmekte böylece kontrol performansını sürekli iyileştirmeye çalışmaktadır. Bu şekilde kontrol süreci sırasında kendisini devamlı uyarlayan akıllı bir kontrol yapısı elde edilmiştir. Bu çalışma, direksiyon sisteminde sinyal gecikmesi olan otonom bir aracın bir sinir ağının diğer bir sinir ağı ile işbirliği yaparak kontrol edildiği literatürdeki ilk kontrol yapısıdır. [5] veya [7]'de sunulan model öngörülü kontrol yöntemlerinin performansı kontrol ufku, kestirim ufku gibi parametrelerin ve amaç fonksiyonlarının başlangıçtaki seçimlerine bağlıdır. Bunların yanlış seçimi kontrol performansını olumsuz etkileyecektir. Benzer şekilde [3] veya [8] 'de sunulan ve durum-uzay

modeli temelli yöntemlerin performansları da seçilen durum-uzay modellerinin parametrelerinin ve yapılarının doğruluğuna bağlıdır. Bu bildiride önerilen yöntemde ise kestirim ufku, kontrol ufku, amaç fonksiyonu, durum uzay modeli parametreleri ve yapısı gibi seçimler bulunmamaktadır. Kontrol sistemi hem aracın dinamiğini öğrenmekte hem de kontrol süreci boyunca en uygun kontrol girişlerini üretmek için kendini devamlı eğitmektedir. Bu şekilde kendini aracın dinamiğine uyarlayan ve kontrol performansını iyileştirme yönünde devamlı hareket eden akıllı bir kontrol sistemi sunulmuştur.

2. Kontrol Sistemi Yapısı

Kontrol sisteminin yapısı Şekil 1'de görülmektedir. İstenen bir yörünge takibi için gereken referans sapma hızı r_{ref} kontrol sisteminin girişidir. Kontrolör ağın girişi sapma hızı hatası $e_r = r_{ref} - r$ dir. Burada r aracın sapma hızı değeridir. Kontrolör ağın çıkışı ise direksiyon açısı kontrol girişi δ 'dır. Model ağın girişi direksiyon açısı, çıkışı ise ağ tarafından hesaplanan sapma hızı r_{NN} dir. Model ağı, direksiyon girişine karşılık aracın verdiği sapma hızı dinamiğini modellemektedir. Kontrol algoritmasında ilk önce model ağın eğitimi gerçekleştirilir. Böylece bu ağ direksiyon girişlerine karşılık aracın sapma hızı davranışını öğrenmiş olur. Bu ağın eğitimi tamamlandıktan sonra kontrolör ağı aracı kontrol etmeye başlar. Kontrolör ağının görevi aracın referans sapma hızı değerlerini takip edebilmesi için gerekli direksiyon girişlerini üretmektir. Kontrol sürecinde kontrolör ağı aynı zamanda devamlı olarak kendini eğitir. Amaç, kontrolör ağının en uygun kontrol girişlerini üretebilmesi böylece sapma hızı hatasının minimize edilmesidir. Fakat kontrolör ağının çıkışı sapma hızı hatası değil direksiyon girişi olduğu için geri yayılım algoritmasının uygulanabilmesi için model ağın yardımı gerekmektedir. Şekil 1'de kesikli çizgili ok ile temsil edildiği gibi model ağın çıkışındaki sapma hızı hatası geri yayılım ile kontrolör ağının çıkışına getirilir ve buradan da kontrolör ağın ağırlıklarının eğitiminde kullanılır. Bir başka deyişle kontrolör ağı aracı kontrol ederken devamlı olarak eğitilmekte, sapma hızı hatasını minimize etmek için gereken direksiyon komutlarını üretmek için ağırlıkları devamlı olarak model ağın yardımı ile güncellenmektedir. Bu süreçte, yani kontrolör ağı aracı kontrol ederken ve ağırlıkları güncellenirken model ağının ağırlıkları sabit tutulmaktadır. Şekil 1'de ayrıca görüldüğü gibi kontrol edilen araç modelinin direksiyon sisteminde sinyal gecikmesi olduğu durum göz önüne alınmıştır. Bir başka deyişle kontrolör ağının çıkışı olan direksiyon açısı komutu direksiyon eyleyicisinde gecikme ile gerçekleşmektedir.

2.1. Model Ağı

İlk olarak model ağın girişlerinin ne olması gerektiğine karar verilmiştir. Bu amaçla tek izli doğrusal araç modeli ele alınmıştır. Aracın yanal kayma açısı β 'nın küçük değerler aldığı kabul edilerek ihmal edilirse sapma hızı ile araç direksiyon girişi arasında şu bağıntı yazılabilir:

$$\dot{r} \approx \frac{-(C_f l_f^2 + C_r l_r^2)}{I_z u} r + \frac{C_f l_f}{I_z} \delta. \tag{1}$$

Burada C_f ve C_r ön aks ve arka aksta kullanılan doğrusal tekerlek modellerinin çapraz hareket sertliği değerleri, l_f ve l_r aracın ağırlık merkezinin sırasıyla ön aksa ve arka aksa olan mesafeleri, I_z aracın sapma eksenindeki atalet momenti, u ise aracın boylamsal hızıdır. Yukarıdaki denklem birinci mertebe bir diferansiyel denklem olduğu için sapma hızı değerleri, bir önceki sapma hızı değerine ve direksiyon girişine bağlıdır. Bu nedenle sapma hızı ile direksiyon açısı arasındaki dinamiği modellemek için kullanılan model ağın yapısında Şekil 1'de görüldüğü gibi girişler olarak direksiyon açısı ve bir önceki sapma hızı değeri seçilmiştir. Ayrıca birim girişten oluşan bir bias girişi de eklenmiştir. Ağın çıkışı ise hesaplanan sapma hızı değeridir. Model ağın gizli katmanında 9 nöron ve bir adet birim bias giriş bulunmaktadır. Model ağın çıkış katmanı ve gizli katmanındaki ağırlıklar şu şekildedir:

$$\mathbf{W}_0 = \begin{bmatrix} w_{01} \\ w_{02} \\ w_{03} \\ \cdots \\ w_{010} \end{bmatrix}^T, \ \mathbf{W}_1 = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ \vdots & \vdots & \vdots \\ w_{91} & w_{92} & w_{93} \end{bmatrix}. \tag{2}$$

Burada örneğin w_{02} gizli katmanın ikinci nöronundan çıkış katmanına olan ağırlıktır. Yine örneğin w_{32} ağın ikinci girişinden gizli katmandaki üçüncü nörona olan ağırlıktır. Model ağın görevi araç dinamiğini öğrenmektir. Bu ağın eğitimi belirli periyotlarla tekrarlanacak, bir başka deyişle zaman içinde araç dinamiğinde oluşabilecek değişimlere karşı devamlı güncel tutulacaktır. Örneğin zaman içinde aracın parametrelerinde değişiklik gerçekleşebilir ve/veya araç dinamiğinde boşluk, ölü-bölge gibi doğrusal olmayan etkiler oluşabilir veya eyleyici/algılayıcılarda sinyal iletişim gecikmeleri ortaya çıkabilir. Model ağı periyodik olarak aracın davranışını modellemek üzere eğitileceği için devamlı aracın güncel davranış karakteristiğini temsil edecek, her zaman aracın dinamik davranışının gerçekçi bir modeli olması sağlanacaktır. Eğitim prosedürü Şekil 2'de görülmektedir. Araca bir direksiyon komutu girişi δ uygulanarak manevra yaptırılır, manevra sırasında aracın sapma hızı değerleri ölçülür, direksiyon girişi ve ölçülen sapma hızı değerleri ile bir eğitim kümesi oluşturulur ve bu eğitim kümesi ile model ağı eğitilir. Şekil 2'de model ağın eğitimi eğimli ok ile temsil edilmektedir. Çalışmada manevra olarak araca arka arkaya slalomlar yaptırılmış, bu manevralar sırasında araçtan alınan sapma hızı değerleri ile eğitim kümesi oluşturulmuştur. Araca uygulanan direksiyon girişi grafiği ile araçtan ölçülen sapma hızı r ile model ağın hesapladığı r_{NN} karşılaştırma grafikleri sayfa kısıtından dolayı burada verilememiştir. Ağırlıkların eğitimi için geri yayılım algoritması şu şekildedir:

$$\mathbf{W_{0yeni}} = \mathbf{W_{0eski}} - \eta_N \nabla_{\mathbf{W_0}} \mathbf{E_N}, \tag{3}$$

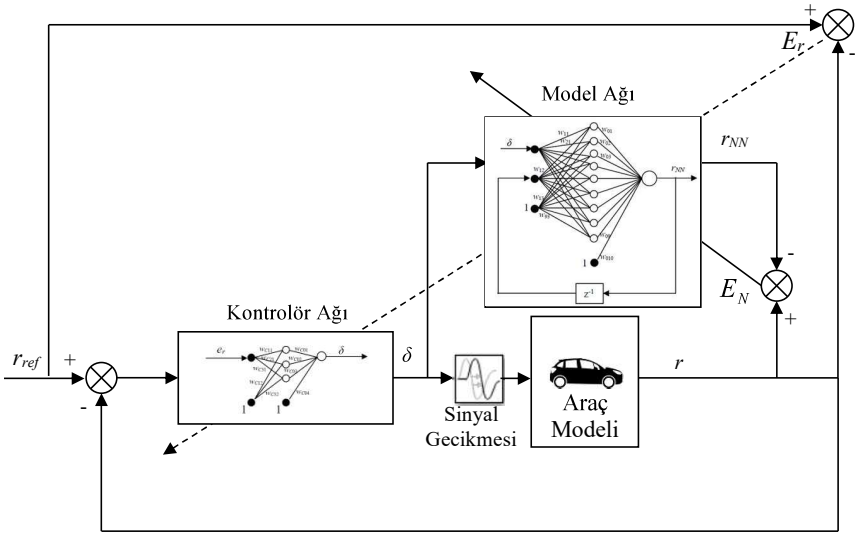
$$\mathbf{W_{1yeni}} = \mathbf{W_{1eski}} - \eta_N \nabla_{\mathbf{W_1}} \mathbf{E_N}. \tag{4}$$

Burada η_N öğrenme adımı büyüklüğü, E_N ise ağın hesapladığı sapma hızı değerinin hata fonksiyonudur ve şu şekilde hesaplanmaktadır:

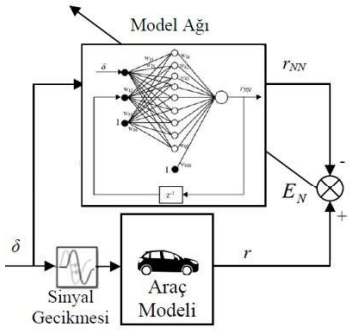
$$E_N = \frac{1}{2} \left(r - r_{NN} \right)^2. \tag{5}$$

Burada r araçtan ölçülen sapma hızı değeri, r_{NN} ise ağ tarafından hesaplanan sapma hızı değeridir. (3) 'te görülen $\nabla_{\mathbf{W_0}} \mathbf{E_N}$ vektörünün elemanları şu şekildedir:

$$\nabla_{\mathbf{W_0}} \mathbf{E_N} = \left[\frac{\partial E_N}{\partial w_{01}}, \frac{\partial E_N}{\partial w_{02}}, \frac{\partial E_N}{\partial w_{03}}, \cdots, \frac{\partial E_N}{\partial w_{010}} \right]. \tag{6}$$



Şekil 1: Kontrol sistemi yapısı.



Şekil 2: Model ağı eğitim prosedürü.

Buradaki kısmı türevler şu şekilde hesaplanmaktadır (i=1,2, ...9),

$$\frac{\partial E_N}{\partial w_{0i}} = -y_{1i}(r - r_{NN}), \tag{7}$$

ve son ağırlık için,

$$\frac{\partial E_N}{\partial w_{010}} = -(r - r_{NN}). \tag{8}$$

Bu ifadelerde y_{1i} değerleri ağın gizli katmanındaki nöronların çıkış değerleridir. Örneğin y_{13} gizli katmandaki üçüncü nöronun çıkış değeridir. Nöronlarda lineer aktivasyon fonksiyonları kullanılmıştır. (4) 'te görülen $\nabla_{\mathbf{W_1}}\mathbf{E_N}$ matrisi elemanları ise şu şekildedir:

$$\nabla_{\mathbf{W_1}}\mathbf{E_N} = \begin{bmatrix} \frac{\partial E_N}{\partial w_{11}} & \frac{\partial E_N}{\partial w_{12}} & \frac{\partial E_N}{\partial w_{13}} \\ \frac{\partial E_N}{\partial w_{21}} & \frac{\partial E_N}{\partial w_{22}} & \frac{\partial E_N}{\partial w_{23}} \\ \vdots & \vdots & \vdots \\ \frac{\partial E_N}{\partial w_{91}} & \frac{\partial E_N}{\partial w_{92}} & \frac{\partial E_N}{\partial w_{93}} \end{bmatrix}. \tag{9}$$

Buradaki kısmı türev değerleri şu şekilde hesaplanmaktadır (i=1,2, ...9),

$$\frac{\partial E_N}{\partial w_{i1}} = -x_1 w_{0i}(r - r_{NN}), \tag{10}$$

$$\frac{\partial E_N}{\partial w_{i2}} = -x_2 w_{0i}(r - r_{NN}), \tag{11}$$

ve,

$$\frac{\partial E_N}{\partial w_{i3}} = -w_{0i}(r - r_{NN}). \tag{12}$$

Bu ifadelerde x_1 ağın giriş katmanındaki birinci giriş, x_2 ise ikinci giriştir. Şekil 1'de görüldüğü gibi birinci giriş direksiyon açısı, ikinci giriş ise hesaplanan sapma hızı değerinin bir önceki değeridir.

2.2. Kontrolör Ağı

Kontrolör ağın girişi sapma hızı hatasıdır. Girişte ayrıca bir adet birim bias giriş te bulunmaktadır. Ağın çıkışı araca uygulanması gereken direksiyon açısı kontrol girişidir. Kontrolör ağı gizli katmanında üç nöron ve bir adet birim bias giriş bulunmaktadır. Ağın çıkış ve gizli katmanındaki ağırlıklar şu şekildedir:

$$\mathbf{W}_{C0} = \begin{bmatrix} w_{C01} \\ w_{C02} \\ w_{C03} \\ w_{C04} \end{bmatrix}^T, \ \mathbf{W}_{C1} = \begin{bmatrix} w_{C11} & w_{C12} \\ w_{C21} & w_{C22} \\ w_{C31} & w_{C32} \end{bmatrix}. \tag{13}$$

Örneğin w_{C03} ağın gizli katmanındaki üçüncü nörondan çıkış katmanına olan ağırlık, w_{C21} ağın birinci girişinden gizli katmandaki ikinci nörona olan ağırlıktır. Kontrolör ağı kontrol süreci boyunca aynı zamanda devamlı eğitim altında olacaktır. Bir başka deyişle aracın referans sapma hızı değerlerini minimum hata ile takip edebilmesi için gerekli araç direksiyon komutu girişlerini hesaplamak için ağırlıkları devamlı eğitilmektedir. Ağırlıkların eğitimindeki kriter sapma hızı hatasının minimum olmasının sağlanmasıdır. Bu nedenle ağırlıkların eğitiminde minimize edilmeye çalışılan fonksiyon şu şekildedir:

$$E_r = \frac{1}{2}\left(r_{ref} - r\right)^2. \tag{14}$$

Bu fonksiyon ile ağırlıklar eğitilecektir fakat sapma hızı hatası ağın çıkışındaki değer değildir. Kontrolör ağın çıkışı direksiyon açısı olduğu için model ağın çıkışındaki E_r değeri geri

yayılım ile kontrolör ağına getirilir ve kontrolör ağının ağırlıkları bu şekilde E_r fonksiyonunu minimize edecek şekilde güncellenir. Ağırlıkların güncellenme kuralları şu şekildedir:

$$\mathbf{W_{C0yeni}} = \mathbf{W_{C0eski}} - \eta_C \nabla_{\mathbf{W_{C0}}} \mathbf{E_r}, \tag{15}$$

$$\mathbf{W_{C1yeni}} = \mathbf{W_{C1eski}} - \eta_C \nabla_{\mathbf{W_{C1}}} \mathbf{E_r}. \tag{16}$$

Burada η_C kontrolör ağı öğrenme adımı büyüklüğüdür. (15)'teki $\nabla_{\mathbf{W_{C0}}} \mathbf{E_r}$ vektörü elemanları şu şekildedir:

$$\nabla_{\mathbf{W_{C0}}} \mathbf{E_r} = \left[\frac{\partial E_r}{\partial w_{C01}}, \frac{\partial E_r}{\partial w_{C02}}, \frac{\partial E_r}{\partial w_{C03}}, \frac{\partial E_r}{\partial w_{C04}} \right]. \tag{17}$$

Buradaki elemanlar şu şekilde hesaplanır ($m = 1, 2, 3$):

$$\frac{\partial E_r}{\partial w_{C0m}} = -\sum_{i=1}^{9} y_{C1m} w_{i1} w_{0i} (r_{ref} - r), \tag{18}$$

$$\frac{\partial E_r}{\partial w_{C04}} = -\sum_{i=1}^{9} w_{i1} w_{0i} (r_{ref} - r). \tag{19}$$

Bu ifadelerde y_{C1m} kontrolör ağının gizli katmanındaki m'nci nöronun çıkış değeri, w_{i1} model ağın birinci girişinden gizli katmanındaki i'nci nöronuna olan ağırlık değeri, w_{0i} ise model ağın çıkış katmanındaki i'nci ağırlık değeridir. Gizli katmandaki ağırlıklar için ise hata gradyeni şu şekilde yazılır:

$$\nabla_{\mathbf{W_{C1}}} \mathbf{E_r} = \begin{bmatrix} \frac{\partial E_r}{\partial w_{C11}} & \frac{\partial E_r}{\partial w_{C12}} \\ \frac{\partial E_r}{\partial w_{C21}} & \frac{\partial E_r}{\partial w_{C22}} \\ \frac{\partial E_r}{\partial w_{C31}} & \frac{\partial E_r}{\partial w_{C32}} \end{bmatrix}. \tag{20}$$

Matrisin elemanları şu şekilde hesaplanır ($m = 1, 2, 3$):

$$\frac{\partial E_r}{\partial w_{Cm1}} = -e_r \sum_{i=1}^{9} w_{C0m} w_{i1} w_{0i} (r_{ref} - r), \tag{21}$$

$$\frac{\partial E_r}{\partial w_{Cm2}} = -\sum_{i=1}^{9} w_{C0m} w_{i1} w_{0i} (r_{ref} - r). \tag{22}$$

Kontrolör ağı aracı kontrol ederken aynı zamanda (15) ve (16) kuralları ile ağırlıkları her bir örnekleme adımında güncellenir, bir başka deyişle kontrol performansını iyileştirmek için ağ devamlı kendini uyarlar.

3. Benzetim Çalışması

Sunulan kontrol algoritmasının benzetim çalışması ile performansı analiz edilmiştir. Şekil 1 'deki araç modeli bloğu olarak çift izli, süspansiyon içeren bir tam araç modeli kullanılmıştır. Araç gövdesi akslar üzerinde yalpa, yunuslama ve dikey hareket serbestlik derecelerine sahiptir. Araç modelinde boylamsal ve yanal tekerlek kuvvetlerinin hesaplanmasında Pacejka tekerlek modeli kullanılmıştır. Araç modelinin detayları [10] 'da görülebilir. Araç kontrol edilirken manevralar sırasında hız düşüşünü engellemek için PID tipi bir hız kontrolcüsü de modele eklenmiştir. Bu hız kontrolcüsü tekerleklere tahrik momentlerini üretmektedir. Benzetim çalışmasında aracın sinüzoidal bir yörüngeyi takip etme senaryosu gerçekleştirilmiştir. Yol profili şu şekildedir:

$$y(x) = M \sin(\frac{2\pi}{s} x). \tag{23}$$

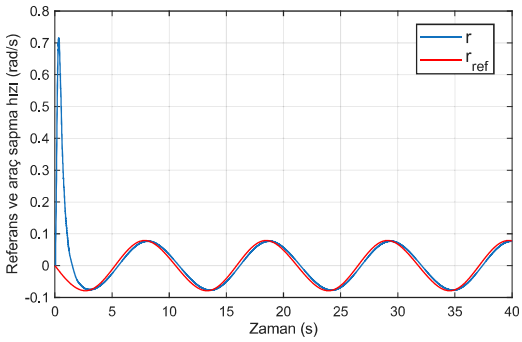
Burada x boylamsal konum, y yanal konum, M sinusoidal yol profilinin genliği, s yol profilinin uzunluğudur. Benzetim çalışmasında sayısal değerler şu şekilde alınmıştır: M= 2 m, s= 100 m. Aracın yol profilini takip etmesi için gerekli referans sapma hızı değerleri şu şekilde hesaplanır:

$$r_{ref} = u \cdot \rho. \tag{24}$$

Burada u boylamsal hız, ρ yörüngenin eğriliğidir ve şu şekilde hesaplanmaktadır:

$$\rho = \frac{y''}{(1 + y'^2)^{3/2}}. \tag{25}$$

Burada, $y' = M\frac{2\pi}{s}\cos(\frac{2\pi}{s}x)$, $y'' = -M(\frac{2\pi}{s})^2 \sin(\frac{2\pi}{s}x)$ dir. (23) 'te verilen yol profilini takip etmek için gerekli referans sapma hızı (24)'ten hesaplandıktan sonra kontrol sistemi bu referans sapma hızını takip edecek şekilde aracı kontrol etmeye başlar. Kontrol sisteminin performansı Şekil 3 'te görülmektedir. Direksiyon sisteminde 50 ms sinyal gecikmesi olduğu durum ele alınmıştır. Benzetim çalışmalarında örnekleme zamanı olarak 0.01 s seçilmiştir.

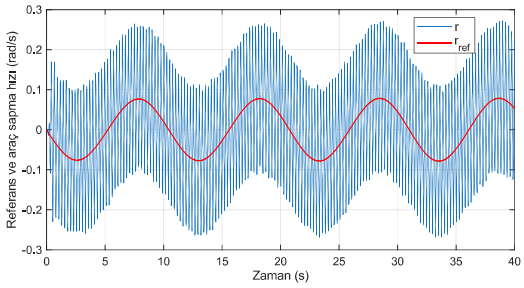


Şekil 3: Önerilen kontrol sisteminin 50 ms sinyal gecikmesi olduğu durumda performansı.

Şekil 3 'te mavi eğri aracın sapma hızı, kırmızı eğri ise referans sapma hızı değerleridir. Görüldüğü gibi kontrolör yapay sinir ağının ağırlıkları başta rastgele atandığı için ağ uygun direksiyon girişi üretememekte ve aracın sapma hızı ile referans sapma hızı arasında ilk başta fark olmaktadır. Fakat ağ hızlı bir şekilde kendini eğiterek referans sapma hızını takip etmek için gerekli direksiyon girişlerini doğru şekilde üretmeye başlamaktadır. Şekil 3 'te görülen kontrollü aracın referans sapma hızı izleme hatası verileri şu şekildedir: Ortalama mutlak hata MAE=0.0201, ortalama karesel hata MSE=0.0055, kök ortalama karesel hata RMSE=0.0741.

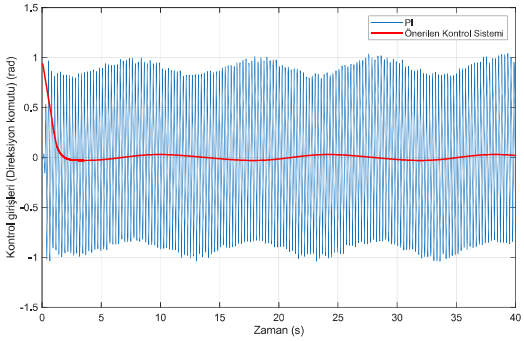
Karşılaştırma için araç modeli bir PI tipi direksiyon kontrolcüsü ile de kontrol edilmiştir. PI kontrolcüsünün girişi sapma hızı hatası $r_{ref} - r$, çıkışı ise direksiyon açısı komutu δ 'dır. PI kontrolcüsünün parametreleri şu şekilde ayarlanmıştır: Direksiyon sisteminde sinyal gecikmesi olmayan bir araç modeli ve MATLAB-Simulink 'teki hazır PID bloğu ile kontrol sistemi kurulmuştur. PID bloğunun "Tune" özelliği ile başlangıç parametre değerleri hesaplatılmış, daha sonra ince ayar ile aracın referans sapma hızını hatasız takip etmesi için gerekli katsayılar belirlenmiştir. Kontrolör parametre değerleri K_p=5 ve K_i=30 'dur. Daha sonra direksiyon sisteminde 50 ms gecikme

olduğu durum için PI direksiyon kontrolcüsünün performansı test edilmiştir. Bu benzetim çalışmasındaki amaç sinyal gecikmesi olmadığı durumda iyi performans gösteren PI direksiyon kontrolcüsünün zaman içerisinde aracın direksiyon sisteminde sinyal gecikmesi oluşması durumunda performansının nasıl değişeceğini gözlemlemektir. PI direksiyon kontrolcüsü sistemde gecikme olmadığı durumda iyi performans gösterirken Şekil 4 'te görüldüğü gibi sinyal gecikmesi olduğu durumda yetersiz kalmaktadır. Önerilen kontrol algoritması ise kendini uyarlamakta, bir başka deyişle sistemi nasıl kontrol etmesi gerektiğini kendi kendine öğrenmektedir.



Şekil 4: PI kontrolörün 50 ms sinyal gecikmesi olduğu durumda performansı.

PI kontrol sistemi ve önerilen kontrol sisteminin kontrol girişleri yani direksiyon açısı komutu değerleri Şekil 5' te görülmektedir. Görüldüğü gibi PI kontrolcüsü kabul edilemez direksiyon girişleri üretirken önerilen kontrolcü kendini uyarladıktan sonra düşük genlikli ve salınımsız direksiyon girişleri ile aracı kontrol etmektedir.



Şekil 5: PI ve önerilen kontrol sisteminin 50 ms sinyal gecikmesi olduğu durumda kontrol girişleri.

4. Sonuçlar

Bu çalışmada birbirleri ile işbirliği yapan iki yapay sinir ağından oluşan bir direksiyon kontrol sistemi sunulmuştur. Ağlardan birincisi sistem dinamiğini öğrenmekte, ikinci ağ ise birinci ağ yardımı ile kontrol süreci boyunca kendini devamlı güncelleyerek kontrol performansını iyileştirmektedir. Bu şekilde kendini devamlı uyarlayan, sistemde oluşabilecek doğrusal olmayan etkilere, bozuculara, sinyal gecikmelerine dayanıklı, kendi kendine öğrenen bir akıllı kontrol sistemi sunulmuştur. Benzetim çalışmasında görülmüştür ki bir PI tipi kontrolcü, aracın direksiyon eyleyicisinde sinyal gecikmesi olması durumunda sistemi kontrol edemez iken önerilen kontrol algoritması kendini devamlı eğiterek aracı nasıl kontrol etmesi gerektiğini kendi kendine öğrenmektedir.

5. Kaynakça

[1] Shuo Cheng, Liang Li, Xiang Chen, Jian Wu, et al. Model-predictive-control-based path tracking controller of autonomous vehicle considering parametric uncertainties and velocity-varying. *IEEE Transactions on Industrial Electronics*, 68(9):8698–8707, 2020.

[2] Xuefang Li, Chengyuan Liu, Boli Chen, and Jingjing Jiang. Robust adaptive learning-based path tracking control of autonomous vehicles under uncertain driving environments. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[3] Xingyu Zhou, Zejiang Wang, Heran Shen, and Junmin Wang. Robust adaptive path-tracking control of autonomous ground vehicles with considerations of steering system backlash. *IEEE Transactions on Intelligent Vehicles*, 2022.

[4] Lubna Khasawneh and Manohar Das. A robust electric power steering angle controller for autonomous vehicles with disturbance rejection. *Electronics*, 11(9):1337, 2022.

[5] Zhongkai Luan, Jinning Zhang, Wanzhong Zhao, and Chunyan Wang. Trajectory tracking control of autonomous vehicle with random network delay. *IEEE transactions on vehicular technology*, 69(8):8140–8150, 2020.

[6] Jie Zhang, Bangji Zhang, Nong Zhang, Chenyang Wang, and Yuanchang Chen. A novel robust event-triggered fault tolerant automatic steering control approach of autonomous land vehicles under in-vehicle network delay. *International Journal of Robust and Nonlinear Control*, 31(7):2436–2464, 2021.

[7] Jiaxing Yu, Xuexun Guo, Xiaofei Pei, Zhenfu Chen, Wei Zhou, Maolin Zhu, and Chaoxian Wu. Path tracking control based on tube MPC and time delay motion prediction. *IET Intelligent Transport Systems*, 14(1):1–12, 2020.

[8] Matteo Corno, Giulio Panzani, Federico Roselli, Michele Giorelli, Davide Azzolini, and Sergio M Savaresi. An LPV approach to autonomous vehicle path tracking in the presence of steering actuation nonlinearities. *IEEE Transactions on Control Systems Technology*, 29(4):1766–1774, 2020.

[9] Raghavendra M Shet, Girish V Lakhekar, Nalini C Iyer, and Sandeep D Hanwate. Robust fuzzy quasi-SMC-based steering control of autonomous vehicle subject to parametric uncertainties and disturbances. *International Journal of Automotive Technology*, pages 1–19, 2024.

[10] Erkin Dinçmen. Neural network steering control algorithm for autonomous ground vehicles having signal time delay. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 238(4):720–736, 2024.

EK - Kontrol Sistemi Kararlılık Analizi

Bu bölümde kontrolör ağı için Lyapunov kararlılık analizi sunulmuştur. Model ağın öğrenme süreci için kararlılık analizi sayfa kısıtından dolayı verilmemiştir. Kararlılık analizi ile (14)'te verilen E_r fonksiyonunun sıfıra yakınsaması şartı irdelenmiştir. Bu amaçla ayrık zamanlı Lyapunov fonksiyonu şu şekilde yazılmıştır:

$$V(k) = \frac{1}{2}\left(r_{ref}(k) - r(k)\right)^2.$$

Burada k örnekleme anı, $V(k)$ ise k'ncı adımdaki Lyapunov fonksiyonu değeridir. Lyapunov fonksiyonundaki değişim şu şekilde formüle edilebilir:

$$\Delta V(k) = \Delta e_r(k) e_r(k) + \frac{1}{2}\Delta e_r^2(k). \tag{26}$$

Bu noktadan itibaren k parametresi denklemlerde görünüş sadeliği açısından gösterilmemiştir. Δe_r şu şekilde elde edilebilir:

$$\Delta e_r = \frac{\partial e_r}{\partial W_{C0}}\Delta W_{C0} + \frac{\partial e_r}{\partial W_{C1}}\Delta W_{C1}, \tag{27}$$

Kontrolör ağı aracı kontrol ettiği için bu ağın ağırlıkları aracın çıkışını belirlemektedir. Bu nedenle (27) ifadesi (26)'ya yerleştirilirse,

$$\begin{aligned}\Delta V &= \frac{\partial e_r}{\partial W_{C0}}\Delta W_{C0} e_r + \frac{\partial e_r}{\partial W_{C1}}\Delta W_{C1} e_r \\ &+ \frac{1}{2}\left(\frac{\partial e_r}{\partial W_{C0}}\Delta W_{C0}\right)^2 + \frac{1}{2}\left(\frac{\partial e_r}{\partial W_{C1}}\Delta W_{C1}\right)^2 \\ &+ \frac{\partial e_r}{\partial W_{C0}}\Delta W_{C0}\frac{\partial e_r}{\partial W_{C1}}\Delta W_{C1},\end{aligned} \tag{28}$$

elde edilir. Ayrıca $\frac{\partial E_r}{\partial W_{C0}} = \frac{\partial e_r}{\partial W_{C0}}\frac{\partial E_r}{\partial e_r}$ ve $\frac{\partial E_r}{\partial W_{C1}} = \frac{\partial e_r}{\partial W_{C1}}\frac{\partial E_r}{\partial e_r}$ ifadeleri kullanılarak şu yazılabilir,

$$\begin{aligned}\frac{\partial e_r}{\partial W_{C0}} &= \frac{1}{e_r}\frac{\partial E_r}{\partial W_{C0}}, \\ \frac{\partial e_r}{\partial W_{C1}} &= \frac{1}{e_r}\frac{\partial E_r}{\partial W_{C1}}.\end{aligned} \tag{29}$$

(29) denklemi (28) denklemine eklenerek ve eğitim kuralları (15) ve (16) numaralı denklemler gözönüne alınırsa,

$$\begin{aligned}\Delta V &= \frac{1}{e_r}\frac{\partial E_r}{\partial W_{C0}}\left(-\eta_C \frac{\partial E_r}{\partial W_{C0}}^T\right) e_r \\ &+ \frac{1}{2}\left[\frac{1}{e_r}\frac{\partial E_r}{\partial W_{C0}}\left(-\eta_C \frac{\partial E_r}{\partial W_{C0}}^T\right)\right]^2 \\ &+ \frac{1}{e_r}\frac{\partial E_r}{\partial W_{C1}}\left(-\eta_C \frac{\partial E_r}{\partial W_{C1}}^T\right) e_r \\ &+ \frac{1}{2}\left[\frac{1}{e_r}\frac{\partial E_r}{\partial W_{C1}}\left(-\eta_C \frac{\partial E_r}{\partial W_{C1}}^T\right)\right]^2 \\ &+ \frac{1}{e_r}\frac{\partial E_r}{\partial W_{C0}}\left(-\eta_C \frac{\partial E_r}{\partial W_{C0}}^T\right) \\ &\cdot \frac{1}{e_r}\frac{\partial E_r}{\partial W_{C1}}\left(-\eta_C \frac{\partial E_r}{\partial W_{C1}}^T\right),\end{aligned} \tag{30}$$

yazılır. Burada,

$$\frac{\partial E_r}{\partial W_{C0}} = \left[\frac{\partial E_r}{\partial w_{C01}}, \frac{\partial E_r}{\partial w_{C02}}, \frac{\partial E_r}{\partial w_{C03}}, \frac{\partial E_r}{\partial w_{04}}\right],$$

ve,

$$\frac{\partial E_r}{\partial W_{C1}} = \left[\frac{\partial E_r}{\partial w_{C11}}, \frac{\partial E_r}{\partial w_{C12}}, \cdots, \frac{\partial E_r}{\partial w_{C32}}\right],$$

şeklindedir. (30) numaralı denklemden aşağıdaki eşitlik yazılabilir:

$$\begin{aligned}\Delta V = &- \eta_C \left\|\frac{\partial E_r}{\partial W_{C0}}\right\|^2 + \frac{1}{2}\frac{1}{e_r^2}\eta_C^2 \left\|\frac{\partial E_r}{\partial W_{C0}}\right\|^4 \\ &- \eta_C \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^2 + \frac{1}{2}\frac{1}{e_r^2}\eta_C^2 \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^4 \\ &+ \frac{1}{e_r^2}\eta_C^2 \left\|\frac{\partial E_r}{\partial W_{C0}}\right\|^2 \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^2.\end{aligned} \tag{31}$$

(31) ifadesi düzenlenerek,

$$\begin{aligned}\Delta V = &-\eta_C \left\|\frac{\partial E_r}{\partial W_{C0}}\right\|^2 \\ &\cdot \left[1 - \frac{1}{2e_r^2}\eta_C \left(\left\|\frac{\partial E_r}{\partial W_{C0}}\right\|^2 + 2\left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^2\right)\right] \\ &-\eta_C \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^2 \left[1 - \frac{1}{2e_r^2}\eta_C \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^2\right],\end{aligned} \tag{32}$$

elde edilir. Gradyenler için üst sınırlar $\left\|\frac{\partial E_r}{\partial W_{C0}}\right\| \leq \left\|\frac{\partial E_r}{\partial W_{C0}}\right\|_{max}$ ve $\left\|\frac{\partial E_r}{\partial W_{C1}}\right\| \leq \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|_{max}$ şeklinde ifade edilirse,

$$\begin{aligned}\Delta V \leq &-\eta_C \left\|\frac{\partial E_r}{\partial W_{C0}}\right\|^2 \\ &\cdot \left[1 - \frac{1}{2e_r^2}\eta_C \left(\left\|\frac{\partial E_r}{\partial W_{C0}}\right\|_{max}^2 + 2\left\|\frac{\partial E_r}{\partial W_{C1}}\right\|_{max}^2\right)\right] \\ &-\eta_C \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|^2 \left[1 - \frac{1}{2e_r^2}\eta_C \left\|\frac{\partial E_r}{\partial W_{C1}}\right\|_{max}^2\right]\end{aligned} \tag{33}$$

eşitsizliğine ulaşılır. Burada ΔV ifadesinin negatif olması için aşağıdaki koşulun sağlanması gerekmektedir:

$$\eta_C < \frac{2e_r^2}{\left(\left\|\frac{\partial E_r}{\partial W_{C0}}\right\|_{max}^2 + 2\left\|\frac{\partial E_r}{\partial W_{C1}}\right\|_{max}^2\right)} \tag{34}$$

Gradyan matrisleri $\partial E_r/\partial W_{C0}$ ve $\partial E_r/\partial W_{C1}$ 'nin elemanları daha önce (18), (19), (21) ve (22) numaralı denklemlerde verilmişti. Kontrolör ağı için (34)'te görülen matris normları için üst sınır değerleri tahmin edilebilir ve böylece kontrolör ağının öğrenme adımı büyüklüğü η_C değeri (34) şartını sağlayacak şekilde seçilirse ΔV negatif olur ve bunun da sonucu E_r 'nin sıfıra yakınsaması garanti edilmiş olunur.