

SOFT HESAPLAMA METOTLARI İLE MOBİL ROBOTLARIN YÖRÜNGELERİNİN HESAPLANMASI

Serkan AYDIN¹ Hakan TEMELTAŞ²

¹ TMYO Endüstriyel Elektronik Programı,
Celal Bayar Üniversitesi, Belediye Fidanlık Tesisleri 45400, Turgutlu, Manisa

² Elektronik ve Haberleşme Mühendisliği Bölümü
Elektrik-Elektronik Fakültesi
İstanbul Teknik Üniversitesi, 80626, Maslak, İstanbul

¹e-posta: serkan.aydin@bayar.edu.tr ² temeltas@ elk.itu.edu.tr

Anahtar Sözcükler: Mobil robot, optimizasyon, yörünge planlama, bulanık mantık, yapay sinir ağları.

ÖZET

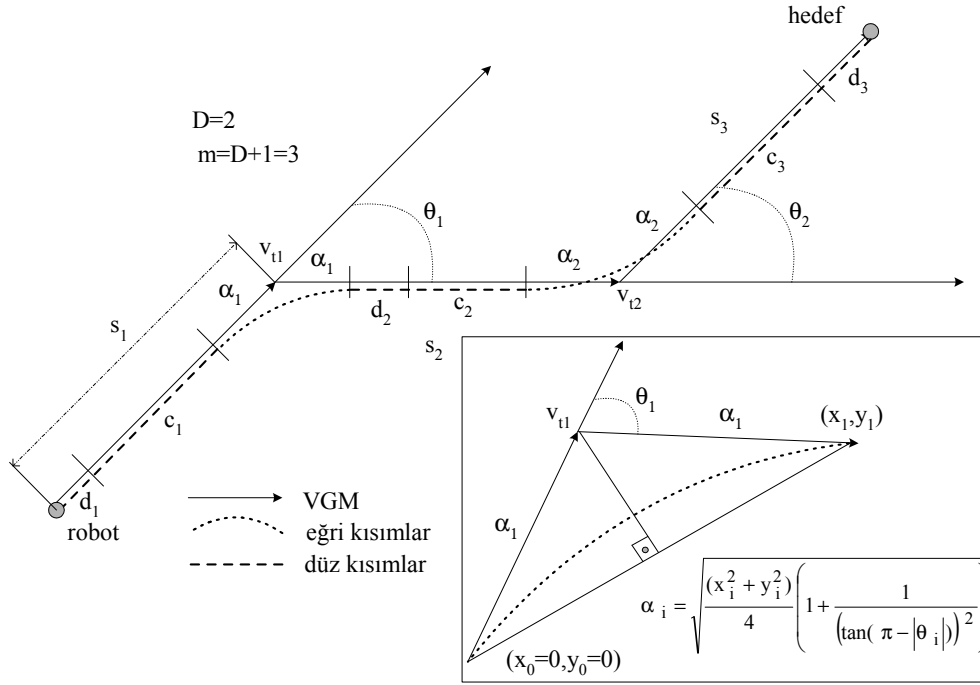
Mobil robotlarda, çevreden gelen kısıtlamalar (engeller) ve robotun hız ve ivme gibi fiziksel kısıtlamaları dikkate alınarak optimal yörünge planlamaları yapılabilmektedir. Genellikle global optimum çözümlere hesaplama sürelerinin çok uzun olduğu nümerik/soft metotlarla ulaşılabilmektedir. Uzun hesaplama sürelerini kısaltmak amacıyla da soft hesaplama teknikleri ön plana çıkmaktadır. Bazı çalışmalarda, önceden görünebilirlik grafik metoduyla (VGM) planlanmış yol üzerinde zaman-optimal yörüngeler bulunmuştur. Yörüngeler, düz kısımlardan (dönme ivmesi $a_s=0$ ve dönme hızı $v_s=0$) ve eğri kısımlardan (öteleme ivmesi $a_t=0$ ve öteleme hızı $v_t=sabit$) oluşturulmuşlardır. Eğri kısımları oluşturmak için sadece θ (dönüş açısı) ve v_t (eğri kısım öteleme hızı) parametrelerinin bilinmesi yeterli olmaktadır. Olası tüm eğri kısımlar kümesi ($\theta \in (0, \pi]^\circ$ ve $v_t \in [0, 40]$ inç) içerisinde global optimizasyon metotları ile, engellerle çakışmayan zaman-optimal amaç ölçütünü sağlayan yörünge seçilmesi oldukça uzun sürede gerçekleştirilmektedir. Bu sürenin kısaltılması için eğri kısımların modellerinin hesaplanmasında yapay sinir ağları ve bulanık mantık metotları kullanılarak deterministik çözüm ile kıyaslandığında daha kısa sürelerde hesaplamalar gerçekleştirilmiştir. Bu çalışmada yapay sinir ağları/ bulanık mantık ile daha doğru modeller gerçekleştirilmiştir. Sonuçlar önceki çalışma sonuçlarıyla kıyaslanmıştır.

1. GİRİŞ

Mobil robotlarda yörünge planlama üzerine son 20 yıldır oldukça yaygın bir şekilde çalışılmaktadır. Literatürdeki bu çalışmaları: 1) Kapalı planlamalar (gerçek zamanda, lokal) ve 2) Açık planlamalar (off-line, global) şeklinde sınıflandırılabilir [1]. Planlama, mobil robot hareket halindeyken yapılıyorsa buna kapalı planlama

denmektedir. Bu tür planlamalarda amaç ölçütümüz sadece hedefe varmaktan ibarettir. Diğer bir deyişle, kapalı planlamalarda, planlanan yörünge istenilen özellikleri sağlaması (minimum mesafe, minimum zaman vb.) her zaman mümkün değildir. Bu şartlar ancak açık planlamalarla yerine getirilebilir. Genellikle açık planlamalarda yörünge: yol planlama, planlanan yolu düzgünleştirme ve düzgünleştirilmiş yolu izleme olmak üzere 3 aşamalı bir hiyerarşiye sahiptirler. Açık planlamaları da, açık-ayrık ve açık-sürekli olmak üzere iki kısma ayırmak mümkündür. Açık-ayrık planlamalarda bu 3 aşama sırasıyla gerçekleşmektedir. Önce ayrıklaştırılmış uzaya (hücrelere ayırma, görünebilirlik grafik metodu vb.) yol planlanır, daha sonra çeşitli şekillerde planlanan yol düzgünleştirilir. 2. aşamada literatürde, B-spline eğrileri, kübik spiraller, polinomal eğriler ve klothoidler [2,3,4,5] gibi değişik eğri aileleri kullanılmıştır. Son aşamada da bu düzgünleştirilmiş yolu izleyecek bir kontrolöre ihtiyaç duyulmaktadır. Bu çalışmalarda genellikle minimum mesafe olarak alınan amaç ölçütü 1. aşamada kullanıldığından dolayı sağlanamayabilmektedir. Diğer bir deyişle global bir amaç ölçütü olmamaktadır. Açık sürekli planlamalarda ise yörünge planlama 2 aşamada tamamlanabilmektedir. İlk aşamada çalışma uzayı ayrıklaştırılmadan planlama yapılır ve sonraki aşamada da bu planlanan yörünge bir kontrolör ile izlenir. Bu kısımdaki çalışmalarda da genellikle varyasyonel metotlar kullanılmıştır. Lagrange çarpanları metodu ile oluşturulan denklem sistemi nümerik metotlarla çözülmüştür.

Bir diğer çalışmada ise [6] yörünge düz kısımlar (dönme hızı $v_s=0$) ve eğri kısımlardan (öteleme hızı $v_t=sabit$) oluştuğu kabul edilmiştir. Düz kısımlarda öteleme hızının artıp azalması maksimum ivme (a_t) ile gerçekleştirilmektedir. Eğri parçacıklarında dönme hızının artış ve azalışları da maksimum ivme (a_s) ile gerçekleştirilirken öteleme hızı sabit alınmaktadır. En son



Şekil 1. s_1 , s_2 ve s_3 doğrusal yol parçacıkları; α_1 α_2 dönüşleri v_{11} v_{12} öteleme hızlarıyla yapabilmek için gereken mesafe; d_1 ise 0'dan v_{11} 'e, d_2 ise v_{11} 'den v_{12} 'ye hız değişim mesafeleri.

olarak *diferansiyel evrim* (DE) ile, serbest konfigürasyon uzayında ($C_{serbest}$), düz kısım ve eğrilerden oluşan, global zaman-optimum amaç ölçütünü sağlayan yörüngeler elde edilmiştir. Uzun olan diferansiyel evrim çözüm süresini kısaltmak amacıyla eğri kısımların hesaplanmasında deterministik model (Robotun denklemlerinin Mclaren seri açılımıyla) yerine :

- Yapay sinir ağı (YSA) öğrenme yoluyla
- Ve Bulanık mantık (BM) metodu

olmak üzere iki değişik şekilde gerçekleştirilmiştir [6, 7]. YSA/BM modellerinin kullanımı ile optimizasyon süresinin 1/8'e kadar düştüğü gösterilmiştir.

Bu çalışmada yapay sinir ağları/ bulanık mantık ile daha doğru modeller gerçekleştirilmiştir. Sonuçlar önceki çalışma sonuçlarıyla kıyaslanmıştır.

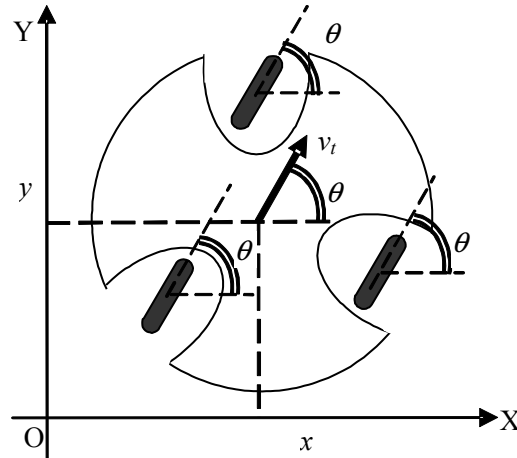
2 YÖRÜNGE PLANLAMA

Mobil robotlar için önceden görünebilirlik grafik metodu (VGM) ile planlanmış yörünge üzerinde çoğul lineer/lineer olmayan kısıtlar altında oluşturulacak izlenebilir yörünge planı bir optimizasyon problemi olarak ele alınamabilmektedir [6]. VGM ile üretilen yol, düz ve eğri kısımlardan oluşmaktadır (Şekil 1).

3 EĞRİ VE DÜZ KISIM DENKLEMLERİ

Tekerlek yapısının Şekil 2'de verildiği, mobil robot (MR) Nomad 200'ün denklemlerinde düz ve eğri kısımların elde edilişi burada incelenmiştir. Robotun hareket esnasında kaymaması :

$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0 \quad (1)$$



Şekil 2. Nomad 200'ün tekerlek yapısı.

nonholonomik kısıtlamasının sağlanmasını gerektirir. Yani verilen bir geometrik konfigürasyon (x, y, θ) altında

anlık öteleme hız vektörünün yönü her zaman tekerlek yönüne paraleldir.

Geometrik konfigürasyona öteleme ve dönme hızlarını ve bunların türevlerini katarak (x, y, θ, v_t, v_s) *dinamik konfigürasyon* olarak isimlendirilmiştir [4]. Burada v_t öteleme hızı, v_s ise dönme (steering velocity) hızıdır. Denklem 1 bu uzayda

$$\begin{aligned}\dot{x} &= v_t \cos(\theta) = v_x \\ \dot{y} &= v_t \sin(\theta) = v_y \\ \dot{\theta} &= v_s \\ \dot{v}_t &= a_t \\ \dot{v}_s &= a_s\end{aligned}\quad (2)$$

şeklinde gösterilebilir (parametrelerin üstündeki nokta zamana göre türev anlamındadır). Burada a_t ve a_s sırasıyla öteleme ve dönme ivmeleridir. Robotun dinamik özelliklerinden dolayı hız ve ivmelerin üzerine:

$$\begin{aligned}|v_t| &\leq v_{t \max} \\ |v_s| &\leq v_{s \max} \\ |a_t| &\leq a_{t \max} \\ |a_s| &\leq a_{s \max}\end{aligned}\quad (3)$$

kısıtlamaları gelmektedir. Uygulamalarda kontrol vektörü $u=[v_t \ v_s]^T$ olarak alınmıştır. Denklem 2'nin her iki tarafının integralini alırsak

$$\begin{aligned}x &= x_0 + \int_{t_0}^t v_t \cos(\theta) dt \\ y &= y_0 + \int_{t_0}^t v_t \sin(\theta) dt \\ \theta &= \theta_0 + \int_{t_0}^t v_s dt \\ v_t &= v_{t0} + \int_{t_0}^t a_t dt \\ v_s &= v_{s0} + \int_{t_0}^t a_s dt\end{aligned}\quad (4)$$

elde edilir. Giriş kısmındaki problem tanımımızdan dolayı Denklem 4'ün son iki eşitliğindeki $a_t=a_{t \max}$ ve $a_s=a_{s \max}$ olarak sabit alındığında :

$$\begin{aligned}v_t &= \left\{ v_{t0} + a_t \Big|_{t_0}^t = v_{t0} + a_t (t - t_0) \right\}_{a_t=a_{t \max}} \\ v_s &= \left\{ v_{s0} + a_s \Big|_{t_0}^t = v_{s0} + a_s (t - t_0) \right\}_{a_s=a_{s \max}}\end{aligned}\quad (5)$$

şeklini alır. Denklem 5'i Denklem 4'ün üçüncü eşitliğinde kullanırsak

$$\begin{aligned}\theta &= \theta_0 + \int_{t_0}^t \left\{ (v_{s0} + a_s (t - t_0)) \right\}_{a_s=a_{s \max}} dt \\ &= \theta_0 + \left\{ (v_{s0} - a_s t_0)(t - t_0) + \frac{a_s (t - t_0)^2}{2} \right\}_{a_s=a_{s \max}}\end{aligned}\quad (6)$$

Denklem 5-6 Denklem 4'ün ilk iki eşitliğinde yerlerine konursa

$$\begin{aligned}x &= x_0 + \int_{t_0}^t \left\{ \begin{aligned} &(v_{t0} + a_t (t - t_0)) \\ &\cos \left(\theta_0 + (v_{s0} - a_s t_0)(t - t_0) + \frac{a_s (t - t_0)^2}{2} \right) \end{aligned} \right\}_{a_s=a_{s \max}, a_t=a_{t \max}} dt\end{aligned}\quad (7)$$

$$y = y_0 +$$

$$\int_{t_0}^t \left\{ \begin{aligned} &(v_{t0} + a_t (t - t_0)) \\ &\sin \left(\theta_0 + (v_{s0} - a_s t_0)(t - t_0) + \frac{a_s (t - t_0)^2}{2} \right) \end{aligned} \right\}_{a_s=a_{s \max}, a_t=a_{t \max}} dt$$

denklemini elde edilir. Denklem 7'deki integrand terimlerinin içerisindeki trigonometrik ifadeler t^2 li ifadeler içerdiğinden dolayı bu integrallere *fresnel* integralleri denir [8]. Belli bir çalışma noktası etrafında trigonometrik terimler

$$\begin{aligned}\cos(\theta) &= \sum_{n=0}^{\infty} (-1)^n \frac{\theta^{2n}}{(2n)!} = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} \dots \\ \sin(\theta) &= \sum_{n=0}^{\infty} (-1)^n \frac{\theta^{2n+1}}{(2n+1)!} = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} \dots\end{aligned}\quad (8)$$

şeklinde Mclaren serilerine açılarak yaklaşık sonuçlar elde edilebilir. Giriş kısmındaki çözüm yaklaşımımızdan hatırlanacağı gibi, yörüngeler düz ve eğri kısımlardan oluşmaktadır. İşlemlerin daha iyi görülebilmesi için, düz kısımlar ve dönüş eğrilerinde, $t_0=0$ alınarak bu kısımlar için Denklem 7 yeniden yapılandırılacaktır.

Düz kısımlarda $a_s=0$ ve $v_{s0}=0$ olduğundan dolayı

$$x = x_0 + \int_0^t \left\{ (v_{t0} + a_t t) \cos(\theta_0) \right\}_{a_t=a_{t \max}} dt$$

$$= \left\{ x_0 + \left(v_{t0} t + a_t \frac{t^2}{2} \right) \cos(\theta_0) \right\}_{a_t=a_{t \max}} \quad (9)$$

$$y = y_0 + \int_0^t \left\{ (v_{t0} + a_t t) \sin(\theta_0) \right\}_{a_t=a_{t \max}} dt$$

$$= \left\{ y_0 + \left(v_{t0} t + a_t \frac{t^2}{2} \right) \sin(\theta_0) \right\}_{a_t=a_{t \max}}$$

denklemleri Denklem 7'den yazılabilir.

Eğri kısımlarda öteleme hızı $v_t=v_{t0}$ olarak sabit olduğundan $a_t=0$ alınır:

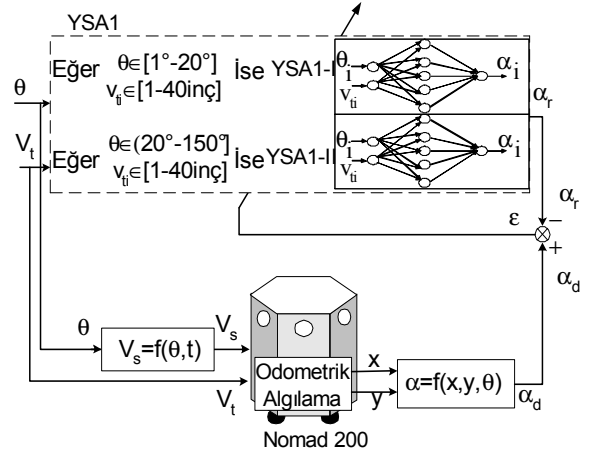
$$x = x_0 + \int_0^t \left\{ \begin{matrix} (v_{t0}) \\ \cos \left(\theta_0 + v_{s0} t + \frac{a_s t^2}{2} \right) \end{matrix} \right\}_{a_s=a_{s \max}} dt \quad (10)$$

$$y = y_0 + \int_0^t \left\{ \begin{matrix} (v_{t0}) \\ \sin \left(\theta_0 + v_{s0} t + \frac{a_s t^2}{2} \right) \end{matrix} \right\}_{a_s=a_{s \max}} dt$$

şeklini alır. Denklem 10, Denklem 8'deki gibi seriye açılabılır. Eğri kısımlar için a_s anahtarlarma süreleri hesaplanır. a_s anahtarlarma sürelerine göre Denklem 10'daki θ_0 , v_{s0} ve $a_{s \max}$ yerlerine konularak x ve y değerleri elde edilmektedir [7].

4 YSA/BM İLE EĞRİ KISIMLARIN MODELLENMESİ

Evolusyoner DE yöntemi ile yapılan optimizasyon işlem süresini azaltmak için yörüngedeki eğri kısımların hesaplanmasında YSA/BM kullanılmış ve sonuçları incelenmiştir. Yörüngelerdeki eğri kısımların modellenmesinde önerilen YSA uygulaması Şekil 3'te verilmiştir. Bu önerilen YSA yapısında, 2 giriş, 5 sinir hücreli 1 gizli katman ve 1 çıkıştan (2:5:1) oluşmaktadır. Bu iki giriş: dönüş açısı ($\theta \in [1^\circ-150^\circ]$) ve dönüşlerdeki öteleme hızı ($v_{ti} \in [1-40 \text{ inç}]$) şeklinde ifade edilir. Çıkış ise, robot doğrusal yol parçacığı üzerinde dönme hareketinin başladığı noktadan itibaren kalan uzunluk miktarı olan α_i 'dir (Şekil 1). Çok katmanlı ileri beslemeli ve hatanın geriye yayılımının kullanıldığı bir ağ mimarisi ile öğrenme işlemi gerçekleştirilmiştir [6].



Şekil 3. YSA ile eğri kısımların modellenmesi

Dönüş açılarına düşen bağıl hataların kendi aralarında sınıflandırılabilmesinden dolayı, YSA yapısında 2 ağ aynı anda eğitilerek aynı hassasiyetteki sonuçlar daha hızlı elde edilmiştir. Böylece YSA yapısı iki ayrı ağdan oluşmuştur. Bunlardan biri, YSA-I ile adlandırılan ve giriş değer aralıkları $\theta \in [1^\circ-20^\circ]$ ve hız $v_{ti} \in [1-40 \text{ inç}]$ olarak belirlenmiştir. Diğer ağ yapısı YSA-II olarak adlandırılmış ve giriş değer aralıkları $\theta \in (20^\circ-150^\circ]$ ve hız $v_{ti} \in [1-40 \text{ inç}]$ olarak verilmiştir. YSA-I ve YSA-II için sırasıyla 154 ve 938 giriş çifti kullanıldı. Yine sırasıyla 3239 ve 20698 çiftle testler gerçekleştirildi. Bu YSA yapılarının hata değerleri Tablo 1'de verilmektedir. Tablo 1'deki karşılaştırmalarda kullanılan hata değerlerini ifade etmek için :

$$E = \{ \epsilon_i \mid \epsilon_i = (\alpha_{di} - \alpha_{ri}) / \alpha_{ri} \quad i \in [1, n] \} \quad (21)$$

bağıl hata kümesini tanımlayalım. Bu hata kümesindeki elemanların ortalaması :

$$E_O = \left(\frac{\sum_{i=1}^n 100|\epsilon_i|}{n} \right) \% \quad (22)$$

$$E_M = \max(|100E|) \% \quad (23)$$

ise hata kümesinin maksimumunu vermektedir. Bağıl hatası %5 ten yüksek olan eleman sayısını E_5 verir:

$$E_5 = \{ \epsilon_i \mid |100\epsilon_i| > 5 \quad i \in [1, n] \} \quad (24)$$

Yörünge hesaplamalarında kullanılan fonksiyonların hesaplanma sürelerinin azaltılabilmesi için bir önceki kısımda önerilen YSA yapılarına alternatif olarak *Bulanık Mantık* (BM) [9] modelleri önerilmiştir. Elde edilen sonuçlar Tablo 1'de verilmiştir. Yukarıda gerçekleştirilen YSA ve BM modellerindeki maksimum hataların en düşüklerinin sırasıyla %7.19 ve

%7.44 olduğu gözlenmektedir. Bu hataları farklı YSA algoritmaları kullanmak suretiyle Tablo 2’ de verildiği gibi %1.7’ ye düşürmüştür. Buna ek olarak YSA-Levenberq-Marquardt 2:5:5:1 mimarisinde %5’ ten büyük hatanın olmadığı da tablodan gözlenmektedir.

Tablo 1. 30000 iterasyon sonrasında YSA-I, YSA-II, ağlarının test fazı hata değerleri.

	Ağ Mim./ Üyelik Fonk	E _O %	E _M %	E _S
YSA1-I	2:5:1	0.88	7.19	6
YSA1-II	2:5:1	0.76	6.22	14
BM Üçgen	2:2	1.49	14.43	295
BM Üçgen	3:3	0.637	10.899	75
BM Yamuk	2:2	1.38	14.72	1106
BM Yamuk	3:3	0.384	7.44	28

Tablo 2. 30000 iterasyon sonrasında farklı ağ yapılarının test fazı hata değerleri.

	2:5:1 Ağ Mimarisi			2:5:5:1 Ağ Mimarisi		
	E _O %	E _M %	E _S	E _O %	E _M %	E _S
YSA Levenberg-Marquardt	0.65	12.71	276	0.12	1.7	0
YSA Quasi-Newton	1.32	19.58	374	9	50	15164
YSA Resilient	2.4	32.2	3252	1.17	16.68	666
YSA Ölçekli eşlenik gradiyent (Scaled conjugate gradient)	0.54	9.73	91	0.24	5.53	3

5 SONUÇLAR VE TARTIŞMA

Bu yayında, mobil robot Nomad 200 için görünebilirlik grafik metodu ile planlanan yol üzerinde zaman-optimal yörünge için robotun denklemlerinin Mclaren serisine açılımıyla yapılması gereken hesaplamalar için değişik eğitim yöntemlerine sahip YSA ve BM modelleri kullanılmıştır. Levenberg-Marquardt hatanın geriye yayılımı ve Ölçekli eşlenik gradiyent (Scaled conjugate gradient) eğitim metodu YSA modellerinin çok daha iyi sonuçlar verdiği görülmüştür.

KAYNAKLAR

- [1] Zefran M., 1996. Continuous Methods for Motion Planning, PhD Thesis, University of Pennsylvania, Philadelphia.
- [2] Zhang J., Knoll A.(1995). An Enhanced Optimization Approach for Generating Smooth Robot Trajectories in the Presence of Obstacles. Proc. Of the European Chinese Automation Conference, London.
- [3] Weber H., (2000). A motion planning and execution system for mobile robots driven by stepping motors, Robotics and Autonomous Systems, **33**, 207–221.
- [4] Reuter J., (1998). Mobile robots trajectories with continuously differentiable curvature: An optimal control approach, IEEE/RSJ Proc. Of Int. Conf on Intelligent Rob. and Sys., Victoria B.C., Canada.
- [5] Scheuer A., Xie M., (1999). Continuous-Curvature Trajectory Planning for Manoeuvrable non-Holonomic Robots, IEEE/RSJ Proc. Of Int. Conf on Intelligent

Robots and System. [6] Aydın S., (2003). Mobil Robotlarda Evrimsel Metotlar ile Hareket Planlama, Doktora Tezi, Fen Bilimleri Enstitüsü, İstanbul Teknik Üniversitesi.

- [7] Aydın Serkan, Temeltaş Hakan, “Planning Optimal Trajectories for Mobile Robots Using an Evolutionary Method with Fuzzy Components”, Advances in Natural Computation, Eds by L. Wang, K. Chen and Y.S. Ong, Springer, LNCS 3612, ISSN 0302-9743, pp. 703-712, 2005
- [8] Kreyszig E., (1993). Advanced Engineering Mathematics. John Wiley& Sons, inc., Singapore.
- [9] Jang R. J. -S., Sun C.-T., Mizutani E., 1997. Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence, Prentice Hall, USA