**Kablosuz Sensör Ağlarında Bulanık Mantık ile Düğüm Lokalizasyonu**

**BTÜ Bilgisayar Mühendisliği Bölümü**

**Esnek Hesaplama Dersi 2024-2025 Dönemi**

**1. Giriş**

Kablosuz Sensör Ağları (WSNs), çevresel parametrelerin izlenmesinde yaygın olarak kullanılmakta olup, bu ağlardaki düğümlerin konumlarının doğru bir şekilde tespit edilmesi son derece önemlidir. Düğüm lokalizasyonu, Kablosuz Sensör Ağlarındaki önemli sorunlardan biridir ve bilinmeyen düğümlerin koordinatlarını, çapa düğümler olarak adlandırılan koordinatları bilinen sensörleri kullanarak tahmin edilen bir süreçtir. Bu ağlarda amaç düşük Ortalama Lokalizasyon Hatası (ALE) ile sonuçlanan optimum ağın bulunmasıdır.

Bu çalışmada, WSN ortamında bilinmeyen düğüm konumlarının tahmini için **Mamdani Bulanık Çıkarım Sistemi** tasarlanmış ve değerlendirilmiştir. Gerçekleştirilen çalışmada, farklı üyelik fonksiyonları ve berraklaştırma yöntemlerinin performansları karşılaştırılmış ve en başarılı kombinasyon belirlenmiştir.

**2. Veri Kümesi**

Kullanılan veri seti **UCI Machine Learning Repository** üzerinden sağlanan "Average Localization Error (ALE) in sensor node localization process in WSNs" veri setidir. Veri setinde toplam 107 örnek bulunmaktadır.

Kullanılan sütunlar:

* anchor\_ratio: Çapa düğüm oranı (%)
* trans\_range: Sensör iletim mesafesi
* node\_density: Sensör düğüm yoğunluğu
* iterations: İterasyon sayısı
* ale: Gerçek Average Localization Error (hedef değişken)

sd\_ale sütunu (standart sapma) bu çalışmada kullanılmamıştır.

Veri setinin ilk birkaç satırı aşağıdaki tabloda görülebilir:

| **anchor\_ratio** | **trans\_range** | **node\_density** | **iterations** | **ale** |
| --- | --- | --- | --- | --- |
| 10 | 25 | 100 | 30 | 8.89 |
| ... | ... | ... | ... | ... |

**3. Yöntem**

**3.1. Mamdani Bulanık Çıkarım Sistemi**

Mamdani Bulanık Çıkarım Sistemi, bulanık kontrol sistemlerinde yaygın olarak kullanılan bir çıkarım mekanizmasıdır. Bu çalışmada, dört giriş değişkeni ve bir çıkış değişkeni tanımlanmıştır. Giriş değişkenleri anchor\_ratio, trans\_range, node\_density ve iterations iken çıkış değişkeni ALE'dir.

Her bir giriş ve çıkış değişkeni için üç dilsel terim (low, medium, high) kullanılmıştır. Kurallar uzman bilgisi ve mantık temelli olarak utils/rule\_generator.py modülünde oluşturulmuştur.

**3.2. Üyelik Fonksiyonları**

Bu çalışmada iki farklı tip üyelik fonksiyonu kullanılmıştır:

* **Triangular (Üçgen)**: Üçgen üyelik fonksiyonu, basitliği ve hesaplama kolaylığı nedeniyle tercih edilmiştir.
* **Gaussian (Gauss)**: Gauss üyelik fonksiyonu, daha yumuşak geçişler sağlaması nedeniyle test edilmiştir.

Her değişken için low, medium ve high etiketleri tanımlanmıştır. Üyelik fonksiyonlarının görsel temsillerini aşağıdaki şekillerde görebilirsiniz.

[BU KISMA ÜYELİK FONKSİYONLARI GRAFİĞİ EKLEYİNİZ - ÖRNEĞİN ÜÇGEN VE GAUSS ÜYELİK FONKSİYONLARININ GÖRSELLERİ]

**3.3. Kural Tabanı**

Sistem için oluşturulan kural tabanı örnek kurallar şunlardır:

1. EĞER anchor\_ratio DÜŞÜK VE trans\_range DÜŞÜK VE node\_density DÜŞÜK VE iterations DÜŞÜK İSE ALE YÜKSEK
2. EĞER anchor\_ratio YÜKSEK VE trans\_range YÜKSEK VE node\_density YÜKSEK VE iterations YÜKSEK İSE ALE DÜŞÜK
3. EĞER anchor\_ratio ORTA VE trans\_range ORTA VE node\_density ORTA VE iterations ORTA İSE ALE ORTA
4. EĞER anchor\_ratio DÜŞÜK VE trans\_range YÜKSEK VE node\_density YÜKSEK VE iterations DÜŞÜK İSE ALE ORTA
5. EĞER anchor\_ratio YÜKSEK VE trans\_range DÜŞÜK VE node\_density DÜŞÜK VE iterations YÜKSEK İSE ALE ORTA
6. EĞER anchor\_ratio DÜŞÜK VE trans\_range ORTA VE node\_density ORTA VE iterations ORTA İSE ALE YÜKSEK
7. EĞER anchor\_ratio YÜKSEK VE trans\_range YÜKSEK VE node\_density DÜŞÜK VE iterations ORTA İSE ALE DÜŞÜK

**3.4. Berraklaştırma Yöntemleri**

İki farklı berraklaştırma yöntemi kullanılmıştır:

* **Center of Sums (COS)**: Bu yöntemde, birleştirilmiş bulanık kümelerin ağırlık merkezleri dikkate alınır.

def center\_of\_sums(output\_range, aggregated\_output):

numerator = np.sum(output\_range \* aggregated\_output)

denominator = np.sum(aggregated\_output) + 1e-6

return numerator / denominator

* **Weighted Average (WA)**: Bu yöntemde, her bir bulanık kural çıktısının merkezi ve etkinlik derecesi kullanılarak ağırlıklı ortalama hesaplanır.

def weighted\_average(output\_activations, output\_centers):

numerator = 0.0

denominator = 0.0

for label, activation in output\_activations.items():

numerator += activation \* output\_centers[label]

denominator += activation

if denominator == 0:

return 0

return numerator / denominator

**4. Sonuçlar**

**4.1. Başarı Metrikleri**

Her bir model kombinasyonu için **Mean Absolute Error (MAE)** ve **Root Mean Squared Error (RMSE)** değerleri hesaplanmıştır. Sonuçlar aşağıdaki tabloda özetlenmiştir:

| **Model** | **MAE** | **RMSE** |
| --- | --- | --- |
| triangular + COS | x.xxx | x.xxx |
| triangular + WA | x.xxx | x.xxx |
| gaussian + COS | x.xxx | x.xxx |
| gaussian + WA | x.xxx | x.xxx |

[BU KISMA KENDİ HESAPLADIĞINIZ MAE VE RMSE DEĞERLERİNİ GİRİNİZ]

**4.2. Grafikler**

Aşağıda farklı modellerin tahmin sonuçları ve gerçek değerler arasındaki ilişkiyi gösteren grafikler sunulmuştur:

[BU KISMA "results/plots/triangular\_COS.png" GRAFİĞİNİ EKLEYİNİZ]

[BU KISMA "results/plots/triangular\_WA.png" GRAFİĞİNİ EKLEYİNİZ]

[BU KISMA "results/plots/gaussian\_COS.png" GRAFİĞİNİ EKLEYİNİZ]

[BU KISMA "results/plots/gaussian\_WA.png" GRAFİĞİNİ EKLEYİNİZ]

Ayrıca tüm modellerin MAE ve RMSE değerlerinin karşılaştırıldığı grafikler:

[BU KISMA "results/overall\_comparison\_mae.png" GRAFİĞİNİ EKLEYİNİZ]

[BU KISMA "results/overall\_comparison\_rmse.png" GRAFİĞİNİ EKLEYİNİZ]

**5. Sonuç ve Değerlendirme**

Yapılan deneysel çalışmalar sonucunda, her bir model kombinasyonunun performansı değerlendirilmiştir. Elde edilen bulgulara göre:

1. En düşük MAE değerini [EN İYİ MAE DEĞERİNE SAHİP MODEL] modeli sağlamıştır.
2. En düşük RMSE değerini [EN İYİ RMSE DEĞERİNE SAHİP MODEL] modeli sağlamıştır.
3. Genel olarak [GENEL DEĞERLENDİRMENİZİ BURAYA YAZINIZ - hangisinin daha iyi performans gösterdiği] modeli daha iyi performans göstermiştir.

Bu çalışma ile bulanık mantık tabanlı sistemlerin WSN ortamında düğüm lokalizasyonu için etkili bir yöntem olduğu ortaya konmuştur. Özellikle [EN İYİ ÜYELİK FONKSİYONU] üyelik fonksiyonu ve [EN İYİ BERRAKLAŞTIRMA YÖNTEMİ] berraklaştırma yönteminin kombinasyonu, bu problem için en etkili çözümü sunmaktadır.

**6. Öneriler**

Bu çalışmanın sonuçlarına dayanarak, gelecekteki araştırmalar için aşağıdaki öneriler sunulabilir:

* Daha karmaşık ve gerçekçi kural tabanları oluşturulabilir.
* Kurallar genetik algoritma gibi optimizasyon teknikleri ile otomatik olarak optimize edilebilir.
* Farklı üyelik fonksiyonu türleri ve berraklaştırma yöntemleri denenebilir.
* Daha büyük ve gerçekçi veri setleri ile test edilmesi önerilmektedir.
* Mamdani'nin yanı sıra Sugeno gibi diğer bulanık çıkarım sistemleri ile karşılaştırmalar yapılabilir.

**7. Kaynakça**

1. [UCI Machine Learning Repository - ALE in WSNs](https://archive.ics.uci.edu/dataset/844/average+localization+error+(ale)+in+sensor+node+localization+process+in+wsns)
2. Zadeh, L. A. (1965). Fuzzy sets. Information and control, 8(3), 338-353.
3. Mamdani, E. H., & Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. International journal of man-machine studies, 7(1), 1-13.
4. BTÜ Esnek Hesaplama Dersi Slaytları

**Ek: Öğrenci Bilgileri**

| **Ad Soyad** | **Numara** |
| --- | --- |
| ... | ... |
| ... | ... |