

Yayınla-Abone Ol Tabanlı Dağıtık Sistemlerde Yapısal Etkileşim Örüntülerinin Çizge Tabanlı Statik Analiz ile İncelenmesi

A Graph-Based Static Analysis of Structural Interaction Patterns in Publish-Subscribe Based Distributed Systems

Mustafa Can Çalışkan^{1,2}, İbrahim Onuralp Yiğit^{1,2}, Feza Buzluca²

¹Komuta Kontrol & Savunma Teknolojileri, HAVELSAN, İstanbul, Türkiye

²Bilgisayar ve Bilişim Fakültesi, İstanbul Teknik Üniversitesi, İstanbul, Türkiye

Özet

Yayınla-abone ol mimarileri, çalışma zamanında gevşek bağlanma sağlarken uygulamalar arası etkileşimlerin örtük kalmasına neden olmakta ve mimari analizi güçleştirmektedir. Bu çalışma, bu örtük yapısal örüntüleri ortaya çıkarmak üzere statik analiz destekli, çizge ve kural tabanlı bir yaklaşım sunmaktadır. Mesajlaşma ilişkileri statik analiz ile çıkarılarak çizge tabanlı bir temsil oluşturulmakta; türetilen metrikler görelî ve kural tabanlı biçimde değerlendirilerek yapısal olarak aykırı bileşenler belirlenmektedir. Yaklaşım, HAVELSAN ADVENT Savaş Yönetim Sistemi üzerinde uygulanmış ve bulgular alan uzmanı değerlendirmeleriyle tutarlı bulunmuştur.

Anahtar Kelimeler

Dağıtık Sistemler, Yayınla-Abone Ol Mimarisi, Mimari Analiz, Yapısal Aykırılık, Çizge Tabanlı Temsil, Yapısal Metrikler, Görelî Değerlendirme, Statik Analiz

Abstract

Publish-subscribe architectures, while enabling loose coupling at runtime, obscure application-level interactions and complicate architectural analysis. This study proposes a static-analysis-supported, graph-based and rule-driven approach to reveal such implicit structural patterns. Messaging relationships are extracted via static analysis and modeled as a graph-based representation; derived metrics are evaluated relatively through rule-based schemes to identify structurally atypical components. The approach has been applied to the HAVELSAN ADVENT Combat Management System, where findings were consistent with expert assessments.

Keywords

Distributed Systems, Publish-Subscribe Architecture, Architectural Analysis, Structural Atypicality, Graph-Based Representation, Structural Metrics, Relative Evaluation, Static Analysis

1. Giriş

Yayınla-abone ol (publish-subscribe) mimarisi, modern dağıtık sistemlerde yaygın olarak kullanılmaktadır. Bu mimari, uygulamaların gevşek bağlanmasına olanak tanırken, uygulamalar arası etkileşimlerin örtük hâle gelmesine neden olmakta ve mimari yapının bütüncül analizini zorlaştırmaktadır.

Konular üzerinden kurulan dolaylı iletişim, zamanla fark edilmesi güç yapısal örüntülerin oluşmasına yol açabilmektedir: aşırı merkezi uygulamalar, yoğun kullanılan iletişim kanalları veya belirli çalışma düğümlerinde kümelenen etkileşimler bu tür yoğunlaşmalara örnek gösterilebilir. Bu örtük yapıların ortaya çıkarılmasında statik analiz, çalışma zamanı verilerine bağımlı kalmadan tüm kod tabanı üzerinden kapsamlı bir görünüm sunabilmektedir [1, 2].

Bu çalışma, yayınla-abone ol tabanlı dağıtık sistemlerdeki örtük etkileşimleri statik analiz yoluyla ortaya çıkarmayı ve çizge tabanlı bir temsil üzerinden tanımlanan yapısal metrikler ile kural tabanlı görelî değerlendirme yaklaşımıyla yorumlamayı amaçlamaktadır. Yaklaşım, doğrudan kusur tespiti değil; sistemin kendi iç bağlamına göre alışılmadık yapısal yoğunlaşmaları görünür kılan açıklayıcı bir analiz çerçevesi sunmayı hedeflemektedir.

Geliştirilen yöntem, HAVELSAN tarafından geliştirilen ADVENT Savaş Yönetim Sistemi üzerinde uygulanmış ve bulguların alan uzmanı değerlendirmeleriyle tutarlı olduğu gözlemlenmiştir.

2. İlgili Çalışmalar

Yazılım mimarilerinin yapısal değerlendirmesi, erken dönemlerde sınıf düzeyindeki yapılara odaklanmıştır. Marinescu'nun *Detection Strategy* yaklaşımı, tekil metrikler yerine birden fazla ölçütün birlikte yorumlanmasını önermekle birlikte [3], analiz biriminin sınıflarla sınırlı kalması mimari seviyedeki dolaylı etkileşimlerin ele alınmasını güçleştirmektedir. Dağıtık mimarilerin yaygınlaşmasıyla Taibi ve arkadaşları [4] ile Garcia ve arkadaşları [5], yapısal anomalilerin mimari seviyede de ortaya çıkabildiğini ve sistemin evrimi üzerinde belirleyici etkiler oluşturabildiğini göstermiştir. Yayınla-abone ol mimarilerinde etkileşimlerin örtük doğası, bu örüntülerin sınıf temelli analizlerle ortaya çıkarılmasını daha da zorlaştırmaktadır [6].

2.1. Bu Çalışmanın Konumu

Bu çalışma, sınıf seviyesindeki ölçüt temelli yorumlama yaklaşımlarının kavramsal temelini koruyarak, analiz birimini uygulama, konu, çalışma düğümü ve ortak kütüphane seviyelerine taşımaktadır. Amaç, doğrudan kusur tespiti değil; mimari yapının kendi iç bağlamına göre görelî karakteristiğini ortaya koyan açıklayıcı bir analiz çerçevesi sunmaktır.

3. Geliştirilen Yaklaşım

Bu çalışmada, yayınla-abone ol tabanlı dağıtık sistemlerdeki örtük etkileşimleri incelemek üzere statik analiz ve kural tabanlı ölçüt yorumlamasına dayanan bir yaklaşım geliştirilmiştir. Temel felsefe, çalışma zamanı verilerine bağımlı kalmadan, kaynak kod ve yapılandırma dosyalarından yola çıkarak örtük etkileşimleri görünür kılmaktır. Yaklaşım üç adımdan oluşmaktadır:

1. **Statik analiz ile ilişkilerin çıkarılması:** Uygulamaların konular üzerinden gerçekleştirdiği yayınlama ve abonelik çağrıları kaynak koddan statik analiz yoluyla tespit edilir.
2. **Çizge temsili üzerinde ölçütlerin hesaplanması:** Etkileşimler yönlü ve etiketli bir çizge üzerinde modellenir; bu temsilden yapısal ölçütler hesaplanır.
3. **Kural tabanlı görelî değerlendirme:** Ölçütler, sistem içerisindeki görelî dağılımları dikkate alınarak yapısal örüntüler oluşturacak şekilde birlikte değerlendirilir.

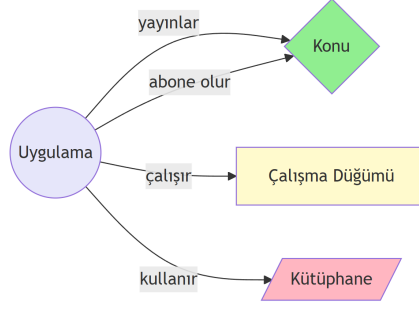
Bu çalışmada *kural* terimi, normatif bir yargıyı değil; ölçütlerin birlikte oluşturduğu yapısal örüntüleri tanımlayan ölçüm temelli biçimsel karar koşullarını ifade etmektedir.

3.1. Çizge Tabanlı Temsil

Önerilen yaklaşımda, yayınla-abone ol tabanlı dağıtık sistem mimarileri yönlü ve etiketli bir çizge temsili kullanılarak modellenmektedir. Bu temsil, Yiğit ve Buzluca tarafından önerilen çok katmanlı mimari etkileşim modelini [7] temel almakta olup ortak kütüphane bağımlılıklarını içerecek biçimde genişletilmiştir. Model; uygulamalar, konular, çalışma düğümleri ve ortak kütüphaneler arasındaki ilişkilerin tekil bir yapı üzerinde ele alınmasını mümkün kılmakta ve yapısal metriklerin çıkarımı için analitik bir temel sunmaktadır.

Şekil 1'de örnek bir çizge temsili sunulmaktadır. Çizge aşağıdaki düğüm türlerini içermektedir:

- *Uygulama düğümleri*, belirli bir işlevi yerine getiren bağımsız uygulamaları,
- *Çalışma düğümü düğümleri*, uygulamaların üzerinde çalıştığı fiziksel veya sanal sunucuları,
- *Konu düğümleri*, uygulamalar arası asenkron iletişimi sağlayan yayın kanallarını,
- *Kütüphane düğümleri*, birden fazla uygulama tarafından ortak kullanılan yazılım bileşenlerini



Şekil 1: Uygulamalar, konular, çalışma düğümleri ve ortak kütüphaneler arasındaki etkileşimleri gösteren örnek çizge temsili (temel model için bkz. [7])

temsil etmektedir.

Düğüm arasındaki yönlü kenarlar; yayınlama ve abonelik ilişkilerini, uygulamaların fiziksel yerleşimlerini ve ortak kütüphane kullanım bağımlılıklarını ifade etmektedir. Bu temsil, yayınlama-abone ol mimarilerinde örtük kalan etkileşimleri görünür hâle getirmekte ve sonraki değerlendirme süreçleri için yapısal bir temel oluşturmaktadır.

3.2. Statik Analiz ile Etkileşimlerin Çıkarılması

Yayınlama-abone ol tabanlı dağıtık sistemlerde, çalışma zamanında etkin olmayan iletişim noktalarının modele dâhil edilmesi hatalı bağımlılık temsiline yol açabilmektedir. Bu nedenle, giriş noktası tabanlı statik analiz yaklaşımı benimsenmiş; ana yürütme noktasından ulaşılabilen çağrı zincirleri incelenerek yalnızca erişilebilir yayınlama ve abonelik çağrıları kapsama alınmıştır.

Çağrı grafiği çıkarımında polimorfizm ve geri çağırım mekanizmaları dikkate alınmıştır. Analiz, CodeQL [8] sorguları aracılığıyla gerçekleştirilmiş; elde edilen etkileşimler çizge tabanlı temsilde yönlü kenarların, çalışma düğümü bilgileri ise yapılandırma tanımlarından elde edilerek modele dâhil edilmiştir.

Kod 1’de bu yaklaşımı örnekleyen sadeleştirilmiş bir CodeQL sorgusu verilmiştir.

```
1 import java
2 from Method mainMethod, MethodCall call
3 where
4   mainMethod.getName() = "main" and
5   call = mainMethod.polyCalls().getACall() and
6   call.getTarget().getName().matches("publish%")
7 select call, "Ana yürütme akisinden erişilebilen yayınlama çağrısı"
```

Kod 1: Ana yürütme noktasından erişilebilen publish çağrılarının tespiti için örnek CodeQL sorgusu

3.3. Yapısal Metriklerin Tanımı

Bu bölümde, yayınlama-abone ol tabanlı dağıtık sistem mimarilerinin yapısal özelliklerini nicel olarak ifade etmek üzere tanımlanan metrikler sunulmaktadır. Metrikler, çizge tabanlı temsil üzerinden uygulama, konu, çalışma düğümü ve ortak kütüphane düzeylerinde tanımlanmış olup her biri mimarinin farklı bir yapısal boyutunu yansıtmaktadır. Bazı metriklerin hesaplanmasında, çizge temsili üzerinden türetilen ikincil soyutlamalardan yararlanılmış; bu soyutlamalar ilgili metrik tanımları kapsamında açıklanmaktadır.

3.3.1. Notasyonlar

Bu çalışmada kullanılan temel notasyonlar aşağıda özetlenmiştir.

- $a \in \mathcal{A}$: Sistem içerisindeki bir uygulama
- $t \in \mathcal{T}$: Sistem genelinde tanımlı bir konu

- $n \in \mathcal{N}$: Bir çalışma düğümü
- $l \in \mathcal{L}$: Ortak kullanılan bir kütüphane
- $PUB(a) \subseteq \mathcal{T}$: Uygulama a tarafından yayın yapılan konular kümesi
- $SUB(a) \subseteq \mathcal{T}$: Uygulama a tarafından abone olunan konular kümesi
- $PUB(t) \subseteq \mathcal{A}$: Konu t 'ye yayın yapan uygulamalar kümesi
- $SUB(t) \subseteq \mathcal{A}$: Konu t 'ye abone olan uygulamalar kümesi
- $RUNS(n) \subseteq \mathcal{A}$: Çalışma düğümü n üzerinde konumlanan uygulamalar kümesi
- $USES(a) \subseteq \mathcal{L}$: Uygulama a tarafından kullanılan kütüphaneler kümesi
- $USES(l) \subseteq \mathcal{A}$: Kütüphane l 'yi kullanan uygulamalar kümesi

Bu notasyonlar, mimari yapının çizge tabanlı temsili üzerinden türetilen yapısal ilişkileri ifade etmekte ve aşağıda tanımlanan yapısal metriklerin formülasyonunda kullanılmaktadır.

3.3.2. Uygulama Düzeyinde Yapısal Metrikler

Uygulama düzeyindeki yapısal metrikler, bir uygulamanın mimari yapı içerisindeki etki alanını, etkileşim biçimini ve rol dağılımını farklı boyutlardan karakterize etmeyi amaçlamaktadır.

Etki Alanı (R – Reach) Bir uygulamanın, yayın yaptığı veya abone olduğu konular aracılığıyla *kendisi dışındaki* uygulamalarla kurduğu benzersiz etkileşimlerin toplam sayısı aşağıdaki şekilde tanımlanmıştır:

$$R(a) = |\{a' \in \mathcal{A} \setminus \{a\} \mid (\exists t \in PUB(a) : a' \in SUB(t)) \vee (\exists t \in SUB(a) : a' \in PUB(t))\}|$$

Yoğunlaştırma (AMP – Amplification) Bir uygulamanın sınırlı sayıda yayın kanalı üzerinden ne ölçüde geniş bir etki alanı oluşturduğunu ifade etmektedir:

$$AMP(a) = \frac{R(a)}{|PUB(a)| + 1}$$

Metrik yalnızca yayıncı rolüne odaklandığından, abonelik kanalları hesaplamaya dâhil edilmemiştir.

Rol Asimetrisi (RA – Role Asymmetry) Bir uygulamanın üretici (yayıncı) ve tüketici (abone) rollerindeki dengesizliği aşağıdaki şekilde ifade edilmiştir:

$$RA(a) = \frac{|PUB(a)| - |SUB(a)|}{|PUB(a)| + |SUB(a)| + 1}$$

Bağlam Çeşitliliği (TC – Topic Context Diversity) Bir uygulamanın etkileşimde bulunduğu konuların farklı işlevsel bağlamlara yayılımını ifade eden bağlam çeşitliliği metriği aşağıdaki şekilde tanımlanmıştır:

$$TC(a) = |\{\text{category}(t) \mid t \in PUB(a) \cup SUB(a)\}|$$

Bu çalışmada konu kategorileri, konu isimlendirme şemalarındaki hiyerarşik önekler üzerinden türetilmiştir.

Kütüphane Maruziyeti (LE – Library Exposure) Bir uygulamanın kullandığı ortak kütüphanelerin sayısını ifade eden kütüphane maruziyeti metriği aşağıdaki şekilde tanımlanmıştır:

$$LE(a) = |USES(a)|$$

3.3.3. Konu Düzeyinde Yapısal Metrikler

Kapsayıcılık (C – Coverage) Bir konunun etkileşimde bulunduğu uygulamaların toplam sayısı aşağıdaki gibi tanımlanmıştır:

$$C(t) = |SUB(t)| + |PUB(t)|$$

Dengesizlik (I – Imbalance) Bir konunun yayıncı ve abone dağılımındaki dengesizliği aşağıdaki gibi tanımlanmıştır:

$$I(t) = \frac{||SUB(t)| - |PUB(t)||}{|SUB(t)| + |PUB(t)| + 1}$$

Fiziksel Yayılım (PS – Physical Spread) Bir konunun etkileşimde bulunduğu uygulamaların kaç farklı çalışma düğümüne yayıldığını ifade eden metrik aşağıdaki gibi tanımlanmıştır:

$$PS(t) = |\{n \in \mathcal{N} \mid \exists a \in SUB(t) \cup PUB(t), a \in RUNS(n)\}|$$

Düşük Bağlantılı Uygulama Oranı (LCR – Low Connectivity Ratio) Bir konuya bağlı uygulamalar arasında, sistem genelinde düşük bağlantı çeşitliliğine sahip olanların oranını ifade eden metrik aşağıdaki gibi tanımlanmıştır:

$$LCR(t) = \frac{|\{a \in PUB(t) \cup SUB(t) : |PUB(a) \cup SUB(a)| \leq k\}|}{|PUB(t) \cup SUB(t)| + 1}$$

Burada k parametresi, bir uygulamanın “düşük bağlantılı” kabul edilmesi için toplam konu bağlantı sayısının üst sınırını ifade etmektedir.

3.3.4. Çalışma Düğümü Düzeyinde Yapısal Metrikler

Düğüm Yoğunluğu (ND – Node Density) Bir çalışma düğümü üzerinde konumlanan uygulamaların sayısı aşağıdaki gibi tanımlanmıştır:

$$ND(n) = |RUNS(n)|$$

Düğüm İçi Etkileşim Yoğunluğu (NID – Node Interaction Density) Bir çalışma düğümü üzerinde konumlanan uygulamalar arasındaki mantıksal etkileşim yoğunluğunu ifade etmek üzere, düğüm içi etkileşim yoğunluğu metriği tanımlanmıştır. Bu metrik, aynı çalışma düğümü üzerinde yer alan iki uygulamanın, en az bir konu üzerinden yayınlama–abonelik ilişkisi kurması durumunda etkileşimde kabul edilmesine dayanmaktadır.

İki uygulama a_i ve a_j arasındaki etkileşim ilişkisi aşağıdaki şekilde tanımlanmıştır:

$$a_i \leftrightarrow a_j \iff \exists t \in \mathcal{T} : (a_i \in PUB(t) \wedge a_j \in SUB(t)) \vee (a_j \in PUB(t) \wedge a_i \in SUB(t))$$

Bu tanım doğrultusunda, bir çalışma düğümü n için düğüm içi etkileşim yoğunluğu aşağıdaki şekilde hesaplanmaktadır:

$$NID(n) = |\{(a_i, a_j) \subseteq RUNS(n) \mid a_i \leftrightarrow a_j\}|$$

3.3.5. Kütüphane Düzeyinde Yapısal Metrikler

Kütüphane Yayınlığı (LC – Library Coverage) Bir kütüphaneyi kullanan uygulamaların sayısı aşağıdaki gibi gösterilmiştir:

$$LC(l) = |USES(l)|$$

Kütüphane Yoğunlaşması (LCon – Library Concentration) Bir kütüphanenin kullanımının çalışma düğümleri üzerindeki yoğunlaşma düzeyini ifade eden metrik aşağıdaki gibi belirtilmiştir:

$$LCon(l) = \max_{n \in \mathcal{N}} |RUNS(n) \cap USES(l)|$$

3.4. Göreli Yorumlama ve Kural Tabanlı Birleşik Değerlendirme

Bu bölümde, bir önceki alt bölümde tanımlanan yapısal metrikler, sistem içerisindeki göreli dağılımları dikkate alınarak yorumlanmakta ve kural tabanlı bir değerlendirme yaklaşımı ile birleştirilmektedir. Amaç, tekil metrik değerleri üzerinden doğrudan karar vermek yerine, metriklerin birlikte oluşturduğu yapısal örüntüleri ortaya çıkarmaktır.

3.4.1. Göreli Metrik Yorumlama

Her yapısal metrik, mutlak eşikler yerine sistem genelindeki göreli konumuna göre değerlendirilmiştir. Her metrik M için birinci çeyrek (Q_1) ve üçüncü çeyrek (Q_3) değerleri esas alınarak sınıflandırma yapılmaktadır:

$$\begin{aligned} M(x) \uparrow &\iff M(x) \geq Q_3(M), \\ M(x) \downarrow &\iff M(x) \leq Q_1(M) \end{aligned}$$

Bazı metriklerin ayrık veya sınırlı değer aralığına sahip olması nedeniyle, çeyrek değerlerinin kendileri de uç konumları temsil edebilmektedir. Bu nedenle, Q_1 ve Q_3 'e eşit değerler de sırasıyla görece düşük ve görece yüksek olarak değerlendirilmiştir. $Q_1 = Q_3$ durumunda ise yorumlama yalnızca mutlak uç değerlere (minimum ve maksimum) sahip bileşenler üzerinden gerçekleştirilmektedir.

3.4.2. Yapısal Aykırılık Örüntüleri

Yapısal metriklerin birlikte değerlendirilmesiyle, mimari yapı üzerinde belirli yapısal örüntüler tanımlanmıştır. Bu örüntüler, tekil bir metrikten ziyade, farklı yapısal boyutları temsil eden metriklerin eşzamanlı olarak görece uç değerler alması durumlarını ifade etmektedir.

Uygulama Düzeyi Yapısal Örüntüler

- **Geniş Etki Alanı (Wide Reach):** Bir uygulamanın mimari yapı içerisinde geniş bir etki alanına sahip olması, yüksek etki alanı ve yüksek yoğunlaştırma değerlerinin birlikte gözlemlenmesiyle ifade edilmiştir.

$$R(a) \uparrow \wedge AMP(a) \uparrow \Rightarrow WR(a)$$

- **Rol Dengesizliği (Role Skew):** Bir uygulamanın üretici veya tüketici rollerinden birinde belirgin biçimde yoğunlaşması, rol asimetrisi metriği üzerinden ifade edilmiştir.

$$RA(a) \uparrow \vee RA(a) \downarrow \Rightarrow RS(a)$$

- **Bağlam Yayılımı (Context Spread):** Bir uygulamanın çok sayıda farklı bağlamda etkileşimde bulunması, bağlam çeşitliliği metriğinin görece yüksek olmasıyla tanımlanmıştır.

$$TC(a) \uparrow \Rightarrow CS(a)$$

- **Ortak Bağımlılık Maruziyeti (Shared Dependency Exposure):** Bir uygulamanın çok sayıda ortak kütüphaneye bağımlı olması, kütüphane maruziyeti metriği üzerinden ifade edilmiştir.

$$LE(a) \uparrow \Rightarrow SD(a)$$

Konu Düzeyi Yapısal Örüntüler

- **İletişim Omurgası (Communication Backbone):** Bir konunun mimari yapı içerisinde merkezi bir iletişim rolü üstlenmesi, yüksek kapsayıcılık ve düşük dengesizlik değerlerinin birlikte gözlemlenmesiyle tanımlanmıştır.

$$C(t) \uparrow \wedge I(t) \downarrow \Rightarrow CB(t)$$

- **Yönlü Yoğunlaşma (Directional Concentration):** Bir konunun ağırlıklı olarak yayıncı veya abone yönünde yoğunlaşması, yüksek dengesizlik değeri ile ifade edilmiştir.

$$I(t) \uparrow \Rightarrow DC(t)$$

- **Çevresel Toplayıcı (Peripheral Aggregator):** Bir konunun, sistem genelinde düşük bağlantı çeşitliliğine sahip uygulamaları bir araya toplaması durumudur. Yüksek LCR değerine sahip bir konu, kendisine bağlı uygulamaların büyük çoğunluğunun sınırlı sayıda konuyla etkileşimde bulunduğuna işaret etmekte ve merkezi iletişim yapısına yeterince entegre olmayan bileşenlerin belirli konular etrafında yoğunlaştığını göstermektedir.

$$LCR(t) \uparrow \Rightarrow PA(t)$$

Çalışma Düğümü Düzeyi Yapısal Örüntüler

- **Yoğunlaşmış Etkileşim Kümesi (Interaction Hotspot):** Bir çalışma düğümü üzerinde hem çok sayıda uygulamanın konumlanması hem de uygulamalar arası etkileşimin yoğun olması durumu aşağıdaki şekilde tanımlanmıştır.

$$ND(n) \uparrow \wedge NID(n) \uparrow \Rightarrow IH(n)$$

Kütüphane Düzeyi Yapısal Örüntüler

- **Yaygın Ortak Kütüphane (Widely Used Library):** Bir kütüphanenin çok sayıda uygulama tarafından kullanılması şeklinde ifade edilmiştir.

$$LC(l) \uparrow \Rightarrow WUL(l)$$

- **Yoğunlaşmış Ortak Kütüphane (Concentrated Library):** Bir kütüphanenin kullanımının belirli çalışma düğümlerinde yoğunlaşması şeklinde tanımlanmıştır.

$$LCon(l) \uparrow \Rightarrow CL(l)$$

3.4.3. Birleşik Aykırılık Skoru

Birleşik aykırılık skoru, mimari yapı içerisinde dikkat gerektiren bileşenlerin sınıflandırılmasını değil, *görelî olarak önceliklendirilmesini* amaçlayan bir göstergedir. Tekil bir metrikte gözlemlenen aşırı uç değerler değerlendirilmesini domine etmesi engellenir; birden fazla boyutta eşzamanlı ortaya çıkan aykırılıklar ise daha belirgin biçimde öne çıkarılır.

Skor iki tamamlayıcı bileşenden oluşmaktadır: (i) birden fazla yapısal boyuttaki aykırılık örüntülerini esas alan *örüntü tabanlı aykırılık skoru* ve (ii) tekil metriklerdeki aşırı uç durumların tamamen göz ardı edilmesini engelleyen, ancak etkisi sınırlandırılmış *tek-boyutlu aykırılık katkısı*.

Örüntü Tabanlı Aykırılık Skoru: Bir bileşenin birden fazla yapısal boyutta aykırı davranış sergilemesi, ilgili bileşen türü için tanımlanan örüntüler üzerinden değerlendirilmektedir. Uygulama düzeyi için:

$$OS_A^P(a) = \sum_{p \in \mathcal{P}_A} \frac{1}{|\{a' \in \mathcal{A} \mid p(a')\}|} \cdot \mathbb{I}[p(a)]$$

Burada $\mathcal{P}_A = \{WR, RS, CS, SD\}$ uygulama düzeyindeki yapısal örüntüler kümesini, $\mathbb{I}[\cdot]$ ise örüntünün sağlanması durumunda 1, aksi hâlde 0 değerini alan Iverson bracket notasyonunu ifade etmektedir.

Normalizasyon terimi, yaygın olarak gözlemlenen örüntülerin birleşik skoru yalnızca yaygınlıkları nedeniyle domine etmesini engellemekte; böylece önsel bir önem sıralaması yapılmaksızın farklı örüntülerin görelî etkileri dengeli biçimde yansıtılmaktadır. Aynı ilke doğrultusunda konu, çalışma düğümü ve kütüphane düzeyleri için skorlar sırasıyla OS_T^P , OS_N^P ve OS_L^P şeklinde tanımlanmaktadır.

Tek-Boyutlu Aykırılık Katkısı: Örüntü tabanlı değerlendirme, yalnızca tek bir metrikte aşırı uç davranış sergileyen bileşenlerin sıralamada geri kalmasına yol açabilmektedir. Bu durumların göz ardı edilmemesi amacıyla, tekil metriklerdeki görelî uçluklar için sınırlı bir katkı terimi tanımlanmıştır.

Her metrik M için, bileşenin sistem genelindeki görelî konumunu ifade eden üst kuyruk temelli bir uçluk değeri $u_M(x) \in [0, 1]$ hesaplanmakta ve belirli bir üst sınır τ ile sınırlandırılmaktadır:

$$c_M(x) = \min(u_M(x), \tau)$$

Bir bileşen için tek-boyutlu aykırılık katkısı, ilgili bileşen türüne ait metrikler üzerinden hesaplanmaktadır:

$$UNI(x) = \sum_{M \in \mathcal{M}_x} c_M(x)$$

Burada \mathcal{M}_x , bileşen x için tanımlı metrikler kümesini ifade etmektedir.

Nihai Birleşik Sıralama Skoru: Bir bileşen için nihai birleşik aykırılık skoru, örüntü tabanlı skor ile tek-boyutlu katkının doğrusal birleşimi olarak tanımlanmıştır:

$$Score(x) = OS^P(x) + \lambda \cdot UNI(x)$$

Burada λ , tek-boyutlu katkının etkisini sınırlamak amacıyla küçük bir değeri seçilmiş olup örüntü tabanlı skurun sıralamanın temel belirleyicisi olmasını sağlamaktadır. Birleşik skor, mutlak bir sınıflandırma mekanizması değil; mimari değerlendirmede hangi bileşenlerin öncelikli incelenmesi gerektiğini belirlemeye yönelik görelî bir sıralama aracı olarak kullanılmaktadır.

4. Vaka Çalışması

Geliştirilen yaklaşımın endüstriyel ölçekteki uygulanabilirliğini değerlendirmek amacıyla gerçek bir yazılım sistemi üzerinde vaka çalışması gerçekleştirilmiştir.

4.1. İncelenen Yazılım Sistemi

Analiz kapsamında, HAVELSAN tarafından geliştirilmiş olan Ağ Destekli Veri Entegre Savaş Yönetim Sistemi (ADVENT) ele alınmıştır. ADVENT, modern deniz harekâtlarında farklı platformlardan elde edilen verileri bütünleşik biçimde işleyerek durumsal farkındalık ve komuta-kontrol fonksiyonlarını destekleyen endüstriyel ölçekli bir dağıtık yazılım sistemidir [9].

Uygulamalar arası etkileşim büyük ölçüde yayımla-abone ol temelli asenkron mesajlaşma üzerinden gerçekleştirilmekte olup bu yapı; uygulamalar, konular, çalışma düğümleri ve ortak kütüphaneler arasında çok katmanlı ve örtük etkileşimlerin oluşmasına zemin hazırlamaktadır [9]. Bu yönüyle ADVENT, önerilen yaklaşımın uygulanabilirliğini değerlendirmek için uygun bir vaka çalışması oluşturmaktadır.

4.2. Analiz Kapsamı ve Varsayımlar

Analiz, sistemin tasarım aşamasındaki yapısal niteliklerine odaklanmaktadır. Kapsam, uygulamalar arası yayımla-abone ol ilişkileri ve dağıtım ortamındaki konumlarla sınırlı olup çalışma zamanına özgü metrikler (mesaj trafiği, performans, gecikme vb.) bilinçli olarak kapsam dışı bırakılmıştır.

Çalışma aşağıdaki varsayımlara dayanmaktadır: Sistemde merkezî bir aracı uygulama bulunmadığı kabul edilmiş olup tüm yayınlama ve abonelik ilişkileri doğrudan uygulama düzeyinde değerlendirilmiştir. Buna ek olarak, ölçütlerin yorumlanmasında mutlak eşikler yerine sistemin iç dağılımına dayalı görelî eşikler kullanılmış; bu doğrultuda bulgular kesin mimari kararlar olarak değil, kanıta dayalı göstergeler olarak ele alınmıştır.

Tablo 1

Bileşen türlerine göre uzman değerlendirmesi sonuçları. Precision@ k isabet oranını, nDCG@ k sıralama kalitesini, Fleiss' κ ise uzmanlar arası uyumu ifade etmektedir.

Bileşen Türü	k	Prec@k	nDCG@k	Fleiss' κ
Uygulama	5	0.80	0.82	0.71
	10	0.70	0.74	
Konu	5	0.80	0.85	0.76
	10	0.80	0.81	
Çalışma Düğümü	5	0.60	0.63	0.65
	10	0.60	0.58	
Kütüphane	5	0.80	0.78	0.70
	10	0.70	0.72	

5. Bulgular ve Değerlendirme

Bu bölümde, önerilen yaklaşımın ürettiği birleşik aykırılık sıralamalarının alan uzmanı değerlendirmeleriyle tutarlılığı incelenmektedir. Değerlendirme üç tamamlayıcı ölçüt üzerinden yürütülmüştür: Precision@ k , sisteme ait ilk k sıradaki bileşenler arasındaki isabet oranını; nDCG@ k , sıralamanın kalitesini –özellikle üst sıralardaki hataları daha ağır cezalandırarak–; Fleiss' κ ise uzmanlar arası değerlendirme uyumunu ifade etmektedir.

Analizlerde düşük bağlantılı uygulama eşiği $k = 2$, tek-boyutlu katkı üst sınırı $\tau = 0.30$ ve birleşik skor ağırlık katsayısı $\lambda = 0.30$ olarak belirlenmiştir. Bu değerler, örüntü tabanlı skorun sıralamadaki ağırlığını korurken tekil metriklerdeki uç durumların tamamen göz ardı edilmemesini amaçlayan sezgisel tercihler olarak seçilmiştir.

Sistem tarafından her bileşen türü için üretilen ilk k bileşen, toplam beş alan uzmanına sunulmuş ve her bileşen için yapısal olarak aykırı olup olmadığı değerlendirilmiştir. Bileşen sayısının fazlalığı nedeniyle uzmanlardan bağımsız bir aykırılık listesi istenmemiş; bunun yerine sisteme ait sıralama üzerinden doğrulama yapılmıştır. Bu yaklaşım, çok sayıda bileşen içeren endüstriyel sistemlerde uzmanların gözden kaçırma riskini ortadan kaldırmaktadır. Bir bileşenin en az üç uzman tarafından aykırı bulunması *olumlu* kabul edilmiştir.

Tablo 1'de özetlenen sonuçlara göre uygulama, kütüphane ve konu düzeylerinde Precision@ k ve nDCG@ k değerleri, yaklaşımın uzman değerlendirmeleriyle yüksek oranda örtüştüğünü göstermektedir. Konu düzeyindeki görece yüksek nDCG@ k değerleri, en belirgin yapısal aykırılıkların sıralamada üst konumlara başarıyla yerleştirildiğine işaret etmektedir. Çalışma düğümü düzeyindeki görece düşük tutarlılık ise dağıtım kararlarının yapısal etkileşim örüntülerinden ziyade operasyonel gereksinimler tarafından şekillendirilebilmesiyle ilişkilendirilmektedir. Fleiss' κ değerleri tüm bileşen türlerinde orta ile yüksek düzeyde uzman uyumuna işaret etmekte olup ground truth olarak kullanılan uzman kararlarının güvenilirliğini desteklemektedir.

6. Sonuç

Bu çalışma, yayınlı–abone ol tabanlı dağıtık sistemlerde örtük kalan etkileşim yapılarını incelemek amacıyla statik analiz ve ölçüt temelli bir değerlendirme yaklaşımı sunmuştur. Uygulamalar, konular, çalışma düğümleri ve ortak kütüphaneler bütünsel bir çizge temsili üzerinde modellenerek etkileşim ağının yapısal özellikleri nicel ölçütlerle ifade edilmiştir.

Analizler sonucunda, sistemin kendi iç dinamiklerine göre görece aykırı davranış sergileyen bileşenler belirlenmiştir. Bu bulgular, doğrudan yargılar üretmekten ziyade mimarinin anlaşılması ve evrimi sırasında dikkate alınabilecek yapısal göstergeler olarak ele alınmalıdır. Yaklaşımın endüstriyel ölçekte uygulanabilirliği, alan uzmanı değerlendirmeleriyle incelenmiş ve sonuçların mevcut mimari bilgilerle tutarlı olduğu gözlemlenmiştir.

Gelecek çalışmalarda, kural tabanlı değerlendirmenin alan uzmanı geribildirimleri ve geçmiş proje verileri kullanılarak denetimli öğrenme yaklaşımlarıyla genişletilmesi; çizge tabanlı temsiller üzerinde

Çizge Sinir Ağları (GNN) ile yapısal örüntülerin otomatik öğrenilmesi araştırılabilir. Buna ek olarak, bu çalışmada sezgisel olarak belirlenen k , τ ve λ hiperparametrelerinin farklı sistem ölçekleri ve mimari yapılar üzerindeki duyarlılık analizi ile sistematik kalibrasyonu ele alınabilir. Ayrıca statik modelin çalışma zamanı verileriyle ağırlıklandırılarak bağlama duyarlılığının artırılması hedeflenmektedir.

Teşekkür

Bu çalışma, HAVELSAN A.Ş. ve İstanbul Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü işbirliği ile yürütülmüştür.

Üretken Yapay Zekâ Kullanım Beyanı

Bu çalışmanın hazırlanması sürecinde yazar(lar), dilbilgisi ve yazım kontrolü amacıyla yapay zekâ araçlarından yararlanmıştır. Bu araç(lar)/hizmet(ler) kullanıldıktan sonra, yazar(lar) içeriği gerektiği şekilde gözden geçirip düzenlemiş ve yayının içeriği için tam sorumluluk almaktadır.

Kaynaklar

- [1] D. Ganesan, M. Lindvall, L. Ruley, R. Wiegand, V. Ly, T. Tsui, Architectural analysis of systems based on the publisher-subscriber style, in: Proceedings of the Working Conference on Reverse Engineering (WCRE), IEEE, 2010, pp. 173–182.
- [2] A. Walker, D. Das, T. Cerny, Automated code-smell detection in microservices through static analysis: A case study, Applied Sciences 10 (2020) 7800. doi:10.3390/app10217800.
- [3] R. Marinescu, Detection strategies: Metrics-based rules for detecting design flaws, in: Proceedings of the 20th IEEE International Conference on Software Maintenance (ICSM), IEEE, Chicago, IL, USA, 2004, pp. 350–359.
- [4] D. Taibi, V. Lenarduzzi, C. Pahl, Microservices anti-patterns: A taxonomy, in: Proceedings of the European Conference on Software Architecture Workshops (ECSAW), ACM, 2017, pp. 1–8.
- [5] J. Garcia, D. Popescu, G. Edwards, N. Medvidovic, Identifying architectural bad smells, in: Proceedings of the 13th European Conference on Software Maintenance and Reengineering (CSMR), IEEE, 2009, pp. 255–258.
- [6] Y. Xing, W. Yang, Z. Tu, J. Yin, D. Chu, Mbst: A bad smells testbed for microservice systems, Cluster Computing (2025). doi:10.1007/s10586-025-05641-1.
- [7] I. O. Yigit, F. Buzluca, A graph-based dependency analysis method for identifying critical components in distributed publish-subscribe systems, in: 2025 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE), 2025, pp. 1–9. doi:10.1109/RASSE64831.2025.11315354.
- [8] GitHub, Codeql: Semantic code analysis engine, <https://codeql.github.com/>, 2023. Accessed: 2025-12-27.
- [9] HAVELSAN, Ağ destekli veri entegre savaş yönetim sistemi (advent), <https://www.havelsan.com/tr/cozumler/advent-savas-yonetim-sistemi>, 2025. URL: <https://www.havelsan.com/tr/cozumler/advent-savas-yonetim-sistemi>, erişim tarihi: 2025-12-27.