

Klasik ve Sezgisel Bulanık İkili Karşılaştırma ile Yazılım Geliştirme Projelerinin Maliyet Tahmini: Uygulama Örneği

Ayfer BAŞAR

Endüstri Mühendisliği, İstanbul Teknik Üniversitesi, İstanbul, Türkiye
ayferbasar@gmail.com

(Geliş/Received:01.06.2016; Kabul/Accepted:03.02.2017)

DOI: 10.17671/gazibtd.309269

Özet— Kaynak ve zaman planlamanın başarılı bir şekilde yürütülebilmesi için, yazılım geliştirme faaliyetlerinde maliyet tahminleme çok önemlidir. Maliyeti düşük tahmin edilen projelerde; gereğinden az kaynak atanması, gerçekçi olmayan planlama vb. sorunlar yaşanabilmekteyken tersi durumda kaynaklar israf edilmekte ve müşteri memnuniyeti düşebilmektedir. Tahmin edilen maliyetin doğruluğu, yazılım geliştirme projelerinin doğru bir şekilde önceliklendirilmesine, kaynakların etkin bir şekilde kullanılmasına, değişiklik isteklerinin başarılı bir şekilde yönetilmesine katkı sağlar. Bu nedenle, maliyetin gerçekçi bir şekilde öngörülmesini sağlayacak uygun yöntemin seçimi, tüm proje paydaşları açısından büyük önem arz etmektedir. Bu çalışmada, öncelikle deneyimli uzmanların görüşünden yararlanarak yazılım geliştirme maliyet tahmini için etkin kriterler belirlenmiş ve Klasik İkili Karşılaştırma yöntemi ile kriterlerin önem seviyeleri elde edilmiştir. Yazılım sektöründeki yoğun rekabet ve belirsizlikler, maliyet tahminleme için ihtiyaç duyulan tam ve kesin bilgi akışını zorlaştırmaktadır. Özellikle önceden belirlenmiş ölçekler kullanılarak yapılan değerlendirmelerde, kriterlerin göreceli öneminin ölçekte yer alan değerler ile tam olarak karşılanamaması, bulanık sayılar ile karar vermenin önemini arttırmaktadır. Bu nedenle, uzman görüşü yardımıyla belirlenen kriterlerin önemini tespiti için çalışmada ayrıca Sezgisel (Hesitant) Bulanık İkili Karşılaştırma önerilmiştir. Yöntemlerin etkinliğinin anlaşılması için, Türkiye’de hizmet vermekte olan bir yazılım şirketinin tamamlanmış projelerinin maliyeti tahmin edilmiştir. Gerçekleşen maliyetlere kıyasla, Sezgisel Bulanık İkili Karşılaştırma yönteminin klasik yöntemden daha başarılı sonuçlar verdiği görülmüştür.

Anahtar Kelimeler— yazılım maliyet tahminleme, karar verme, Klasik İkili Karşılaştırma, Sezgisel Bulanık İkili Karşılaştırma, uygulama

Estimating the Cost of the Software Projects with Traditional and Hesitant Fuzzy Pairwise Comparison: An Application

Abstract— Estimating the cost of software is a fundamental process in terms of resource and time planning. While underestimation causes committing fewer resources than the project needs and setting an unrealistic schedule, overestimation induces waste of resources and low customer satisfaction. By the help of efficient estimation methods, software projects can be prioritized efficiently, resources can be used effectively, and change request can be managed appropriately. Thus, choosing the appropriate cost estimation method is crucial for all the project shareholders. This paper presents a Traditional Pairwise Comparison (TPC) by using a number of criteria selected with the help of expert judgments working in banking and information technology industry in Turkey. It is clear that there may be uncertainties while evaluating the relative importance of two factors which may not be solved by TPC. In order to cope with this uncertainty in judgement, Hesitant Fuzzy Pairwise Comparison (HFPC) is also presented to estimate cost of software projects. In order to analyze the efficiency of proposed approaches, they are applied to a software cost estimation problem for a Turkish company. It is seen that HFPC provides more efficient estimations due to low deviation between the real and estimated effort in comparison with TPC.

Keywords— software cost estimation, decision making, Traditional Pairwise Comparison, Hesitant Fuzzy Pairwise Comparison, application.

1. GİRİŞ (INTRODUCTION)

Teknolojik gelişmelerin etkisiyle, yazılım geliştirme projelerinin işletmelerin rekabet gücü üzerindeki etkisi gün geçtikçe artmaktadır. Bu nedenle kurumlar, yazılım projelerinin zaman ve maliyet açısından etkin yönetimine önem vermek zorundadır. Projenin tahmini geliştirme maliyeti; çizelgeleme, kaynak tahsisi, performans yönetimi, müşteri memnuniyeti vb. kriterler açısından tüm paydaşlar açısından kritik öneme sahiptir. Ayrıca tahmin edilen maliyetin doğruluk seviyesi, projelerin öncelik ve planlama bilgisi üzerinde doğrudan belirleyici özelliğe sahiptir. Yazılım geliştirme maliyeti; efor (adam.gün / ay / hafta), süre (gün, ay, hafta) ya da parasal olarak tahmin edilebilmektedir.

Bu çalışmada, yazılım geliştirme projelerinin maliyet tahmini için önerilen yöntemler sunulmuştur. Maliyetin doğru tahmin edilmesi için alanında uzman olan yöneticilerin görüşü ile etkin kriterler belirlendikten sonra, Klasik ve Sezgisel Bulanık İkili Karşılaştırma yöntemleri kullanılarak kriterlerin göreceli önem seviyeleri elde edilmiştir. Elde edilen sonuçlar, gerçek hayat uygulaması üzerinde kıyaslanmış ve Sezgisel Bulanık İkili Karşılaştırma yönteminin daha başarılı sonuçlar verdiği görülmüştür. Makalenin devamı şu şekilde özetlenmiştir: 2. Bölümde maliyet tahminleme yöntemleri ile ilgili literatürde yer alan çalışmalar sunulacak, 3. Bölümde önerilen Klasik ve Sezgisel Bulanık İkili Karşılaştırma yöntemleri detaylandırılacak, 4. Bölümde önerilen yöntemler bir yazılım şirketi üzerinde uygulanacak, 5. Bölümde ise sonuçlar ve gelecek çalışmalar aktarılacaktır.

2. MALİYET TAHMİNLEME YÖNTEMLERİ LİTERATÜR TARAMASI (LITERATURE REVIEW ON COST ESTIMATION METHODS)

Yazılım projelerinin maliyet tahmini için kullanılan yöntemler, temel olarak algoritmik olan ve olmayan şeklinde iki kategoride incelenmektedir. Uzman görüşü, algoritmik olmayan yöntemlerin en yaygın örneğidir. Konuya ilgili deneyimli uzmanların görüşünden yararlanabilme fırsatı, uzman görüşünün en önemli avantajıdır. Ancak zaman zaman uzmanlar tutarsız ya da taraflı davranabilmekte, böylece tahmin edilen maliyetler gerçekçi olmayabilmektedir [1, 2]. Uzman görüşü ile ortaya çıkabilecek tutarsızlıkları önlemek için Delphi Tekniği [3, 4] ve İkili Karşılaştırma [5] da maliyet tahmini için literatürde yaygın kullanılan algoritmik olmayan yöntemler arasında yerini almıştır. Algoritmik olmayan yöntemlerin diğer örneği olan Parkinson yaklaşımında maliyet, mevcut kaynaklar ve teslimat zamanına dayalı olarak öngörülmektedir. Parkinson yönteminde basitçe, teslimat zamanı 12 ay olan bir proje için 5 kişinin çalışması durumunda, çalışmanın maliyeti 60 adam-ay olarak tahmin edilmektedir [6].

Ancak algoritmik olmayan yöntemler ile elde edilen sonuçların, analitik ya da deneysel/matematiksel

modellere dayalı olmaması ve objektiflikten uzaklığı algoritmik yöntemlerin önemini arttırmıştır. Algoritmik yöntemlerde; x_1, x_2, \dots, x_n maliyet kriterleri ve $f(x_1, x_2, \dots, x_n)$ önceden belirlenen maliyet kriterlerinin kullanıldığı fonksiyon yardımıyla yazılım maliyeti tahmin edilir. Dolayısıyla maliyet kriterleri ve kullanılan fonksiyonun uygunluğu, gerçekçi tahmin açısından büyük öneme sahiptir. En yaygın maliyet kriterleri; ürün, bilgisayar, proje ve personeldir. Öte yandan algoritmik yöntemlerde; doğrusal, çarpımsal ve kuvvet şeklinde fonksiyon kullanılabilir. x_1, x_2, \dots, x_n maliyet kriterleri ve a_1, a_2, \dots, a_n ilgili kriterlerin önem düzeyini göstermek koşuluyla, doğrusal modellerde maliyet $a_0 + \sum_{i=1}^n a_i x_i$ formülü ile tahmin edilir [7]. Çarpımsal modelde maliyet tahmini için, $a_0 \prod_{i=1}^n a_i^{x_i}$ formülü kullanılır [8]. Kuvvet fonksiyon modellerinde ise maliyet; $a \times S^b$ formülü ile tahminlenir. Bu formülde, a ve b maliyet kriterleri, S ise kod büyüklüğünü temsil eder. Kuvvet fonksiyonlarının bilinen ve en yaygın örneği Constructive Cost Model (COCOMO) tekniğidir. Bu yöntemde, kod büyüklüğü olarak "satır sayısı" ve efor olarak "adam-ay" bilgisi referans alınır [9]. COCOMO yönteminin en büyük zorluğu, yazılım projesinin teslimatından önce kod satır sayısının doğru ölçülmesidir. Bu nedenle COCOMO yönteminde çoğu zaman, uzman görüşü ile Proje Değerlendirme ve Gözden Geçirme (Project Evaluation and Review Technique - PERT) tekniği kullanılarak kod büyüklüğü tahmin edilir. Böylece, uzmanlar 3 şekilde görüş bildirir: S_l en düşük, S_n en yüksek ve S_m en olası kod büyüklüğünü temsil eder. Daha sonra, $S = (S_l + 4S_m + S_n) / 6$ formülü ile kod büyüklüğü tahmin edilir. COCOMO yöntemi, genel olarak üç gruba ayrılır: i) Basit COCOMO, ii) Orta COCOMO, iii) COCOMO II. Basit COCOMO yöntemi, maliyet tahmini için $a \times (\text{KLOC})^b$ formülü kullanılan en eski COCOMO yaklaşımıdır [9]. Bu formülde, a ve b değerleri kodun karmaşıklığını temsil ederken, KLOC (Kilos of Lines of Code) ise 1.000 satırlık kod anlamına gelmektedir. Basit COCOMO yöntemindeki a ve b değerleri, yazılımın karmaşıklığına göre değişir. Bu parametreler sırasıyla; karmaşıklık düşükse 2,4 ve 1,05; orta derece ise 3,0 ve 1,15; yüksek ise 3,6 ve 1,20 değerini alır. Açıkça görülmektedir ki, basit COCOMO yönteminde maliyet sadece kod büyüklüğüne bağlıdır ve tahmini etkileyen diğer kriterler göz ardı edilmektedir. Bu nedenle, ilgili diğer parametreleri de göz önünde bulunduracak şekilde Orta COCOMO yöntemi geliştirilmiştir. Orta COCOMO yönteminde b parametresinin değerleri Basit COCOMO ile aynı iken; a parametresi değişmekte; ayrıca ürün, bilgisayar, proje ve personel başlığı altında toplam 15 kriterin farklı seviyelerdeki ağırlık değerlerine göre tahmin formülünde kullanılmak üzere Efor Ayar Kriteri (Effort Adjustment Factor) hesaplanmaktadır. Basit ve Orta COCOMO yöntemlerinin en önemli eksiği; yazılım projesinin tek ve homojen bir ürün olduğunu varsaymak, böylece büyük sistemlerin farklı karakterde çok sayıda alt sistem içerebileceğini göz ardı etmektir. Bu nedenle COCOMO II yöntemi geliştirilmiştir [10]. Bu yöntemde, b parametresi yeni 5 faktöre (hazır bulunma, geliştirme

esnekliği, risk yönetimi, takım kaynaşması ve süreç olgunluğu) göre farklı değerler almaktadır.

Son dönemlerde müşteri gereksinimlerindeki belirsizliğin etkisiyle, işletmeler açısından hızlı üretim önem kazanmış ve çevik (agile) yaklaşımlar yaygın hale gelmiştir. Geleneksel yöntemlerden farklı olarak sık aralıklarla müşteriye ürün teslimatına dayalı çevik yaklaşımda, yazılım maliyet tahminleme için biyolojik bir sinir hücrelerinden oluşan paralel bilgi işleme yapıları olan yapay sinir ağları kullanılmıştır [11, 12]. Ayrıca özellikle problem boyutunun büyük olduğu durumlarda, en iyi ya da en iyiye yakın çözümleri bulmak için arama uzayını hızlı bir şekilde araştıran farklı meta sezgisel yöntemler de maliyet tahminleme probleminin çözümü için önerilmiştir. Örneğin Harmoni Arama Algoritması [13], Tabu Arama [14], Genetik Algoritma [14, 15, 16], Yapay Arı Algoritması [15] konuyla ilgili literatürde yer alan örneklerdendir. Harmoni Arama Algoritması, bir orkestrada görev alan müzisyenlerin çaldığı notalar ile armonik açıdan en iyi melodinin elde edilmesine dayalıdır. Tabu Arama yöntemi, yerel en iyi noktaya takılmayı engellemek için mevcut çözümü iyileştirmeyen hareketlere izin vermekte ve aramanın tarihçesini tutan tabu listelerinin kullanımı sayesinde önceki aşamalarda ziyaret edilen noktalara tekrar uğrama olasılığını düşürmektedir. Evrim teorisi esasına göre çalışan Genetik Algoritmada, potansiyel çözümler ikili olan ya da olmayan sistemlere dayalı veri yapısında basit diziler olarak şifrelenmekte ve söz konusu dizilere problemin çözümüne yönelik işlemler uygulanmaktadır. Yapay Arı Algoritması ise arı topluluklarının davranışlarını temel alan algoritmadır. Meta sezgisel yöntemlerin dışında, tek ve kesin değer yerine bir aralık kullanılarak bulanık mantık ile tahminleme yapılan uygulamalar da mevcuttur [17, 18].

3. İKİLİ KARŞILAŞTIRMA İLE KARAR VERME (DECISION MAKING WITH PAIRWISE COMPARISON)

Çok Kriterli Karar Verme (ÇKKV) yöntemleri; genellikle birden fazla sayıda ve çoğu zaman birbiri ile çelişen amaç ve kriterin olduğu, kriterler arasında tercih yapmanın zor olduğu durumlarda kullanılır. Böylece, ÇKKV problemlerinin sonuçları çoğunlukla karar vericilerin tercihlerine dayalıdır [19]. ÇKKV problemlerindeki kriterlerin önem düzeyini belirlemek için; uzman görüşünden yararlanılabilmesi, her defasında sadece iki kriterin karşılaştırılması, kolay ve anlaşılır olması vb. avantajları sayesinde, literatürde etkin bir yöntem olan Klasik İkili Karşılaştırma önerilmiştir [20]. Ancak, karar sürecindeki belirsizlik ve öznellik dolayısıyla Klasik İkili Karşılaştırma yönteminin etkin olmadığı durumlar söz konusu olabilmektedir. Bu nedenle literatürde, Bulanık İkili Karşılaştırma yöntemi önerilmiştir [21]. Bulanık İkili Karşılaştırma, tek ve kesin bir değer yerine bir aralığı kullanarak karar vericilere iyimser-orta-kötümser şeklinde farklı tercih sunma fırsatı sağlar. Ayrıca yakın zamandan

bu yana Sezgisel Bulanık İkili Karşılaştırma da literatürdeki yerini almıştır [22, 23].

2.1. Klasik İkili Karşılaştırma Yöntemi (Traditional Pairwise Comparison Method)

İkili Karşılaştırma, karar vericilerin yargısına göre her kriterin amaca katkısı açısından göreceli öneminin belirlendiği yöntemdir [24]. Öncelikle; uzman görüşünün toplanabilmesi için amaç, ana ve varsa alt kriterleri içeren karar hiyerarşisi hazırlanmaktadır. Toplam kriter sayısının n olduğu durumda, hiyerarşik yapıdaki kriterlerin göreceli ağırlıklarının belirlenmesi için, $n \times n$ boyutundaki karşılaştırma matrisi (A) yardımıyla karar verici tarafından ikili karşılaştırmalar yapılmaktadır. $i=1,2,3,\dots,n$ ve $j=1,2,3,\dots,n$ olmak üzere matris elemanları arasında $a_{ij}=1/a_{ji}$ ilişkisi bulunmakta olup $i=j$ olduğu için matrisin köşegenleri 1 değerini almaktadır.

$$A = \begin{bmatrix} 1 & a_{21} & a_{31} & \dots & a_{n1} \\ 1/a_{21} & 1 & a_{32} & \dots & a_{n2} \\ 1/a_{31} & 1/a_{32} & 1 & \dots & a_{n3} \\ \dots & \dots & \dots & \dots & \dots \\ 1/a_{n1} & 1/a_{n2} & 1/a_{n3} & \dots & 1 \end{bmatrix}_{n \times n} \quad (1)$$

Kriterlerin ikili karşılaştırması için [24] tarafından geliştirilen Tablo 1'deki önem ölçeği kullanılmaktadır.

Tablo 1: Önem Ölçeği [24] (Importance Scale)

| Değer | Tanım |
|------------|---|
| 1 | Elemanlar eşit öneme sahiptir. |
| 3 | 1. eleman 2.'ye göre biraz daha önemli veya biraz daha tercih edilmektedir. |
| 5 | 1. eleman 2.'ye göre fazla önemli veya fazla tercih edilmektedir. |
| 7 | 1. eleman 2.'ye göre çok fazla önemli veya çok fazla tercih edilmektedir. |
| 9 | 1. eleman 2.'ye göre aşırı derecede önemli veya aşırı derecede tercih edilmektedir. |
| 2, 4, 6, 8 | Ara değerler |

İlgili matristeki her bir öğenin diğer öğelere göre önemini gösteren $n \times 1$ boyutundaki öz vektör, (2)'deki formüller aracılığıyla hesaplanır:

$$b_{ij} = a_{ij} / \left(\sum_{i=1}^n a_{ij} \right) \quad w_i = \left(\sum_{j=1}^n b_{ij} \right) / n \quad (2)$$

2.2. Bulanık İkili Karşılaştırma Yöntemi (Hesitant Pairwise Comparison Method)

ÇKKV problemlerindeki belirsizliği önlemek için bulanık küme teorisi önerilmiştir [25]. Bulanık küme teorisinde bir eleman, bir küme yerine k ($0 \leq k \leq 1$) derecesine ve "üyelik fonksiyonu" adındaki aralığa dahil olmaktadır (örn. üçgen üyelik fonksiyonu). Karar vericilerin belirsiz ortamlarda kriterler arasında tercih sunabilmesi için [25] tarafından bulanık ağırlıklandırma yöntemi önerilmiş ve

$a_{ij} = (l_{ij}, m_{ij}, u_{ij})$ şeklinde üçgen üyelik fonksiyonu kullanılmıştır. Bu gösterimde l alt limit, m orta değer ve u ise üst limiti temsil etmektedir. İlk olarak, karar vericiler Tablo 2’de yer alan bulanık ölçeğe uygun olarak tercihlerini belirtmekte, sonra bulanık karşılaştırma matrisi $A = (a_{ij})$ oluşturulmakta, en sonunda ise her kriter ve alternatif için bulanık önem seviyeleri tespit edilmektedir.

Tablo 2: Bulanık İkili Karşılaştırma Yönteminde Kullanılan Önem Ölçeği [26] (Importance Scale Used in Fuzzy Pairwise Comparison)

| Önem | Dilsel Ölçek | Üyelik fonksiyonu | Aralık | Üçgen bulanık ölçek (l, m, u) |
|---|-------------------|--|--|----------------------------------|
| 1 | Eşit önem | $\mu_M(x) = (3-x) / (3-1)$ | $1 \leq x \leq 3$ | (1, 1, 1) (1, 1, 3) |
| 3 | Biraz daha önemli | $\mu_M(x) = (x-1) / (3-1)$ $\mu_M(x) = (5-x) / (5-3)$ | $1 \leq x \leq 3$ $3 \leq x \leq 5$ | (1, 3, 5) |
| 5 | Fazla önemli | $\mu_M(x) = (x-3) / (5-3)$ $\mu_M(x) = (7-x) / (7-5)$ | $3 \leq x \leq 5$ $5 \leq x \leq 7$ | (3, 5, 7) |
| 7 | Çok fazla önemli | $\mu_M(x) = (x-5) / (7-5)$ | $5 \leq x \leq 7$ | (5, 7, 9) |
| 9 | Aşırı önemli | $\mu_M(x) = (9-x) / (9-7)$ $\mu_M(x) = (x-7) / (9-7)$ | $7 \leq x \leq 9$ | (7, 9, 9) |
| Yukarıdaki önem seviyesi, i kriterinin j kriteri ile karşılaştırılması durumunda kullanılır. Aksi durumda, j kriteri tersine değeri alır. | | | | Tersine değer $1/u, 1/m, 1/l$ |

2.2.1. Sezgisel Bulanık Kümeler (Hesitant Fuzzy Sets)

Karar vericiler tarafından kriterlerin göreceli öneminin belirlenmesi sırasında yaşanabilecek kararsızlık, bir elemanın üyelik derecesinin belirlenmesini zorlaştırmaktadır. Bu nedenle, [27] tarafından sezgisel bulanık kümeler tarif edilmiştir. Sezgisel bulanık kümelerin en önemli özellikleri şu şekildedir:

Tanım 1: Eğer X sabit bir küme ise, X 'in bağlı olduğu sezgisel bulanık küme $E = \{ \langle x, h_E(x) \rangle \mid x \in X \}$ ile gösterilir. Burada $h_E(x)$, E kümesindeki $x \in X$ elemanının $[0, 1]$ değerini alan olası üyelik derecesini temsil eder.

Tanım 2: Üst ve alt limitler, sırayla $h^-(x) = \min h(x)$ ve $h^+(x) = \max h(x)$ ile tespit edilir.

Tanım 3: h kümesinin tümleyeni, $h^c = U_{\in h} \{1-\gamma\}$ ile elde edilir. **Tanım 4:** Her biri sezgisel bulanık küme olan h, h_1 ve h_2 ile yapılan temel matematiksel işlemler şu şekildedir:

$$h^2 = U_{\gamma \in h} \{\gamma^2\} \quad (3)$$

$$\lambda h = U_{\gamma \in h} \{1 - (1-\gamma)^\lambda\} \quad (4)$$

$$h_1 \cup h_2 = U_{\gamma_1 \in h_1, \gamma_2 \in h_2} \{\max\{\gamma_1, \gamma_2\}\} \quad (5)$$

$$h_1 \cap h_2 = U_{\gamma_1 \in h_1, \gamma_2 \in h_2} \{\min\{\gamma_1, \gamma_2\}\} \quad (6)$$

$$h_1 \oplus h_2 = U_{\gamma_1 \in h_1, \gamma_2 \in h_2} \{\gamma_1 + \gamma_2 - \gamma_1 \gamma_2\} \quad (7)$$

$$h_1 \otimes h_2 = U_{\gamma_1 \in h_1, \gamma_2 \in h_2} \{\gamma_1 \gamma_2\} \quad (8)$$

Tanım 6: $(a_1, a_2, \dots, a_n) = \sum_{j=1}^n w_j b_j$ formülü aracılığıyla Ordered Weighting Averaging (OWA) operatörü hesaplanır. Bu formülde b_j ; a_1, a_2, \dots, a_n arasındaki j . en büyük değeri ve $[0, 1]$ arasındaki w_i ise i kriterinin önem düzeyini gösterir ($\sum_{j=1}^n w_j = 1$).

Tanım 7: Sezgisel bulanık kümelere bağlı olarak, karşılaştırmalı dilsel ifadeler $\tilde{A} = (a, b, c)$ şeklinde üçgen bulanık üyelik fonksiyon ile temsil edilir. $\tilde{A} = (a, b, c)$, şu şekilde hesaplanır:

$$a = \min \{ a_L^i, a_M^i, a_M^{i+1}, \dots, a_M^j, a_R^j \} = a_L^i \quad (9)$$

$$b = \text{OWA}_w (a_M^i, a_M^{i+1}, \dots, a_M^j) \quad (10)$$

$$c = \max \{ a_L^i, a_M^i, a_M^{i+1}, \dots, a_M^j, a_R^j \} = a_R^j \quad (11)$$

2.2.2. Sezgisel Bulanık İkili Karşılaştırma Yöntemi (Hesitant Fuzzy Pairwise Comparison Method)

Sezgisel Bulanık İkili Karşılaştırma yönteminin adımları şu şekildedir:

Adım 1: Tüm kriter ve alt kriterler için dilsel ölçeklere uygun olarak uzman görüşü toplanır, ikili karşılaştırma matrisleri oluşturulur.

Adım 2: Tablo 3’te yer alan ölçek kullanılarak, dilsel ifadeler üçgen bulanık sayılara dönüştürülür, uzman görüşü yardımıyla ikili karşılaştırma matrisi \tilde{A}^k oluşturulur ve farklı uzmanların görüşlerinin birbiri ile tutarlı olup olmadığı kontrol edilir.

Tablo 3: Sezgisel Bulanık İkili Karşılaştırma Yönteminde Kullanılan Ölçek (Scale Used in Hesitant Fuzzy Pairwise Comparison)

| Dilsel İfade | Sembol | Üçgen Bulanık Sayı |
|-----------------------|--------|--------------------|
| Kesinlikle Çok Yüksek | KÇY | 7,9,9 |
| Çok Yüksek | ÇY | 5,7,9 |
| Aslında Yüksek | AY | 3,5,7 |
| Zayıf Derecede Yüksek | ZDY | 1,3,5 |
| Eşit Derecede Yüksek | EDY | 1,1,3 |
| Tam Eşit | TE | 1,1,1 |
| Eşit Derecede Düşük | EDD | 1/3,1,1 |
| Zayıf Derecede Düşük | ZDD | 1/5,1/3,1 |
| Aslında Düşük | AD | 1/7,1/5,1/3 |
| Çok Düşük | ÇD | 1/9,1/7,1/5 |
| Kesinlikle Çok Düşük | KÇD | 1/9,1/9,1/7 |

Adım 3: Uzman görüşlerinin, [28] tarafından önerilen bulanık yöntemle göre birleştirilmesi amacıyla Tablo 3’te verilen ölçek, s_0 en düşük ve s_g en yüksek aralığı göstermek üzere büyükten küçüğe doğru sıralanır. Böylece uzman görüşünün s_i ve s_j arasında değişmesi halinde, $s_0 \leq s_i < s_j \leq s_g$ ilişkisi geçerli olur. $\tilde{A} = (a, b, c)$ üçgen bulanık üyelik fonksiyonunun a ve c parametreleri (9) ve (11) formüllerinde tarif edildiği şekilde, b

parametresi ise OWA operatörü aracılığıyla şu şekilde hesaplanır:

$$b = \begin{cases} a_m^i & \text{Eğer } i + 1 = j \\ OWA_w(a_m^i, \dots, a_m^j) & \text{Aksi durumda} \end{cases} \quad (12)$$

OWA işleminde ihtiyaç duyulan ağırlık vektörü, [29] tarafından tarif edilmiştir: $[0,1]$ aralığında bir α parametresinin kullanıldığı, g 'nin Tablo 3'teki ölçekte yer alan terim sayısını, j 'nin en yüksek ve i 'nin en düşük değerlendirme değerini gösterdiği durumda $W = (w_1, w_2, \dots, w_n)$ şu şekilde hesaplanır:

$$w_1 = \alpha^{n-1}, w_2 = (1 - \alpha)\alpha^{n-2}, \dots, w_n = (1 - \alpha) \quad (13)$$

$$\alpha = (g-j+i) / (g-1) \quad (14)$$

Adım 4: $\hat{c}_{ij} = (c_{ij}^l, c_{ij}^m, c_{ij}^u)$ değerlerini içeren Ortak İkili Karşılaştırma Matrisi \hat{C} oluşturulur. Üçgen bulanık sayıların tersi, $\hat{c}_{ji} = (1 / c_{ij}^u, 1 / c_{ij}^m, 1 / c_{ij}^l)$ şeklinde hesaplanır.

Adım 5: $\hat{r}_i = (\hat{c}_{i1} \otimes \hat{c}_{i2} \dots \otimes \hat{c}_{in})^{1/n}$ formülü aracılığıyla; Ortak İkili Karşılaştırma Matrisindeki her satırın bulanık geometrik ortalaması tespit edilir.

Adım 6: $\hat{w}_i = \hat{r}_i \otimes (\hat{r}_1 \oplus \hat{r}_2 \dots \oplus \hat{r}_n)^{-1}$ formülü yardımıyla her kriterin bulanık ağırlık değeri elde edilir.

Adım 7: i kriterinin sıralamasının tespit edilmesi için, $D_i = w_{i,l} + 4w_{i,m} + w_{i,u} / 6$ eşitliğinden yola çıkarak üçgen bulanık sayılar, kesin sayılara dönüştürülür.

Adım 8: Bulanık olmayan ağırlık değerleri normalize edilir ve kriterler, kesin ağırlık değerlerine göre sıralanır.

4. KLASİK VE SEZGİSEL BULANIK İKİLİ KARŞILAŞTIRMA YÖNTEMİNİN UYGULANMASI (APPLICATION OF TRADITIONAL AND HESITANT FUZZY PAIRWISE COMPARISON)

Yazılım projelerinin maliyet tahmini için önemli kriterlerin tespit edilmesi amacıyla ilgili literatür detaylı incelenmiş ve uzman görüşünden yararlanmıştır [30, 31, 32, 33]. Böylece, bir yazılım geliştirirken sarf edilen çabanın tamamını kapsayacak şekilde en önemli kriterler geliştirilen ara yüz, uygulama, yığın iş, veri tabanı ve veri nesne sayısı olarak tespit edilmiştir. Uzmanlar ile yapılan yüz yüze görüşmeler ve doldurulan anketler aracılığıyla, maliyeti etkileyen bu kriterlerin bulanık olan ve olmayan ağırlık değerlerinin belirlenmesi için ihtiyaç duyulan bilgi toplanmıştır. Kriterlerin göreceli öneminin tespit edilmesi amacıyla, Türk yazılım sektöründe yönetici pozisyonunda çalışmakta olan 6 uzmanın görüşüne başvurulmuştur. Böylece 6 uzman; ara yüz (f_1), uygulama (f_2), yığın iş (f_3), veri tabanı (f_4) ve veri (f_5) nesne sayılarını temsil eden 5 farklı kriteri öncelikle Tablo 2'de verilen önem ölçüğü ile ikili olarak karşılaştırmıştır. Tablo 4, uzmanların Klasik İkili Karşılaştırma sonuçlarını göstermektedir.

Tablo 4: Maliyet Kriterlerinin Klasik İkili Karşılaştırması (Traditional Pairwise Comparison of Cost Criteria)

| Uzman 1 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | 1 | 3 | 5 | 9 | 9 |
| f_2 | | 1 | 1 | 5 | 7 |
| f_3 | | | 1 | 1 | 5 |
| f_4 | | | | 1 | 5 |
| f_5 | | | | | 1 |

| Uzman 2 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | 1 | 5 | 5 | 1 | 3 |
| f_2 | | 1 | 5 | 1 | 7 |
| f_3 | | | 1 | 1 | 3 |
| f_4 | | | | 1 | 9 |
| f_5 | | | | | 1 |

| Uzman 3 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | 1 | 3 | 3 | 1 | 9 |
| f_2 | | 1 | 7 | 7 | 5 |
| f_3 | | | 1 | 5 | 3 |
| f_4 | | | | 1 | 9 |
| f_5 | | | | | 1 |

| Uzman 4 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | 1 | 5 | 9 | 0,33 | 5 |
| f_2 | | 1 | 5 | 1 | 5 |
| f_3 | | | 1 | 1 | 1 |
| f_4 | | | | 1 | 3 |
| f_5 | | | | | 1 |

| Uzman 5 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | 1 | 2 | 7 | 5 | 9 |
| f_2 | | 1 | 3 | 9 | 3 |
| f_3 | | | 1 | 1 | 1 |
| f_4 | | | | 1 | 7 |
| f_5 | | | | | 1 |

| Uzman 6 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | 1 | 7 | 9 | 3 | 7 |
| f_2 | | 1 | 1 | 0,2 | 2 |
| f_3 | | | 1 | 0,2 | 1 |
| f_4 | | | | 1 | 5 |
| f_5 | | | | | 1 |

İkinci adımda; uzmanların Klasik İkili Karşılaştırma yoluyla bildirdiği görüşlerin geometrik ortalama ile birleştirilmesiyle elde edilen sonuçlar Tablo 5'te özetlenmiştir. Örnek olarak birinci kriterin ikinci kriterle kıyasla ortak ikili karşılaştırma değeri $(3 \times 5 \times 3 \times 5 \times 2 \times 7)^{(1/6)} = 3,83$ olarak hesaplanmıştır.

Tablo 5: Maliyet Kriterlerinin Klasik İkili Karşılaştırma Sonuçlarının Birleştirilmesi (Collaboration of Traditional Pairwise Comparison)

| | f_1 | f_2 | f_3 | f_4 | f_5 |
|-------|-------|-------|-------|-------|-------|
| f_1 | 1,00 | 3,83 | 5,91 | 1,88 | 6,52 |
| f_2 | 0,26 | 1,00 | 2,84 | 1,99 | 4,41 |
| f_3 | 0,17 | 0,35 | 1,00 | 1,00 | 1,89 |
| f_4 | 0,53 | 0,50 | 1,00 | 1,00 | 5,91 |
| f_5 | 0,15 | 0,22 | 0,53 | 0,17 | 1,00 |

Formül (2)'ye göre hesaplanan öz vektör ve kriterlerin ağırlık değerleri ise Tablo 6'da verilmiştir.

Tablo 6: Klasik İkili Karşılaştırma ile Elde Edilen Öz Vektör ve Ağırlık Değerleri (Eigenvector and Importance Weights Found by Traditional Pairwise Comparison)

| | Öz Vektör Değerleri | | | | | Ağırlık Değerleri |
|-------|---------------------|-------|-------|-------|-------|-------------------|
| | f_1 | f_2 | f_3 | f_4 | f_5 | |
| f_1 | 0,47 | 0,65 | 0,52 | 0,31 | 0,33 | 0,46 |
| f_2 | 0,12 | 0,17 | 0,25 | 0,33 | 0,22 | 0,22 |
| f_3 | 0,08 | 0,06 | 0,09 | 0,17 | 0,10 | 0,10 |
| f_4 | 0,25 | 0,08 | 0,09 | 0,17 | 0,30 | 0,18 |
| f_5 | 0,07 | 0,04 | 0,05 | 0,03 | 0,05 | 0,05 |

Tablo 6'deki ağırlık değerleri incelendiğinde, yazılım geliştirme projelerinin maliyet tahmini üzerinde en büyük etkiye sahip kriterin %46'lık pay ile ara yüz ile ilgili nesne sayısını temsil eden kriter (f_1) olduğu görülmektedir. %22 ve %18 oranındaki ağırlığa sahip uygulama (f_2) ve yığın iş nesne sayısı (f_3) kriterlerinin de gerçekçi maliyet tahmini açısından önemli olduğu görülmektedir. Veri tabanı ve veri nesne sayısı ile ilgili kriterler (f_4 ve f_5) ise %10 ve %5 değerindeki ağırlık ile düşük öneme sahiptir.

Klasik İkili Karşılaştırma yönteminin yanı sıra, ayrıca belirsiz ortamlardaki durumun analiz edilmesi için uzmanlar; f_1, f_2, f_3, f_4 ve f_5 kriterlerini Tablo 3'te verilen bulanık dilsel ölçek yardımıyla karşılaştırmıştır. Tablo 7, 6 uzmana ait bulanık ikili karşılaştırma sonuçlarını göstermektedir.

Tablo 7: Maliyet Kriterlerinin Sezgisel Bulanık İkili Karşılaştırması (Hesitant Fuzzy Pairwise Comparison of Cost Criteria)

| Uzman 1 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | TE | EDY | AY | ÇY | KÇY |
| f_2 | | TE | TE | ÇY | ÇY |
| f_3 | | | TE | EDD | ZDY |
| f_4 | | | | TE | ÇY |
| f_5 | | | | | TE |

| Uzman 2 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | TE | ZDY | ZDY | ZDD | EDY |
| f_2 | | TE | ÇY | EDD | AY |
| f_3 | | | TE | EDY | EDY |
| f_4 | | | | TE | KÇY |
| f_5 | | | | | TE |

| Uzman 3 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | TE | ZDY | EDY | TE | KÇY |
| f_2 | | TE | ÇY | ÇY | ÇY |
| f_3 | | | TE | AY | ZDY |
| f_4 | | | | TE | ÇY |
| f_5 | | | | | TE |

| Uzman 4 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | TE | AY | ÇY | AD | ÇY |
| f_2 | | TE | AY | EDY | ZDY |
| f_3 | | | TE | ZDD | TE |
| f_4 | | | | TE | AY |
| f_5 | | | | | TE |

| Uzman 5 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | TE | EDY | ÇY | ZDY | ÇY |
| f_2 | | TE | AY | KÇY | ZDY |
| f_3 | | | TE | ZDD | TE |
| f_4 | | | | TE | AY |
| f_5 | | | | | TE |

| Uzman 6 | f_1 | f_2 | f_3 | f_4 | f_5 |
|---------|-------|-------|-------|-------|-------|
| f_1 | TE | AY | ÇY | ZDY | AY |
| f_2 | | TE | EDD | ZDD | EDY |
| f_3 | | | TE | ÇD | ZDD |
| f_4 | | | | TE | ÇY |
| f_5 | | | | | TE |

Tablo 7'de yer alan değerlendirmeler incelendiğinde uzman görüşlerinin tutarlı olduğu, böylece söz konusu uzmanların görüşlerinden yararlanarak Ortak İkili Karşılaştırma Matrisinin oluşturulabileceği anlaşılmaktadır. Tüm uzmanların kriterleri değerlendirme aralığı, Tablo 8'de verilmiştir.

Tablo 8: Uzmanların Sezgisel Bulanık Değerlendirme Aralıkları (Intervals of Fuzzy Evaluations)

| Uzman | f_1 | f_2 | f_3 | f_4 | f_5 |
|-------|-------|--------------------|--------------------|---------------------|---------------------|
| f_1 | TE | AY ve EDY arasında | ÇY ve EDY arasında | ÇY ve AD arasında | KÇY ve EDY arasında |
| f_2 | | TE | ÇY ve EDD arasında | KÇY ve ZDD arasında | ÇY ve EDY arasında |
| f_3 | | | TE | ÇY ve AD arasında | ZDY ve ZDD arasında |
| f_4 | | | | TE | KÇY ve AY arasında |
| f_5 | | | | | TE |

(9), (10) ve (11) denklemleri aracılığıyla, üçgen bulanık kümeler belirlenmiştir. Tablo 8'de yer alan bulanık değerlendirmeler ve OWA operatörü kullanılarak bulunan üçgen bulanık kümeler Tablo 9'da sunulmaktadır. Örnek olarak; birinci kriterin ikinci ve üçüncü kritere kıyasla bulanık ağırlığının hesaplanma yöntemi şu şekildedir:

Birinci kriterin ikinci kritere kıyasla bulanık değeri AY ve EDY arasında, böylece $i = 6$ ve $j = 8$ değerindedir. (9) ve (11) denklemlerine göre, a ve c parametreleri $\{a_L^i, a_M^i, a_M^{i+1}, \dots, a_M^j, a_R^j\} = \{1, 3, 5, 7\}$ aralığının sırasıyla minimum ve maksimum değerlerini temsil etmektedir. Bu nedenle, $a = 1$ ve $c = 7$ 'dir. b parametresini hesaplamak için;

$$\alpha = (g-j+i) / (g-1) = (10-8+6) / (10-1) = 0,889$$

$$w_1 = \alpha^{2-1} = 0,889 \text{ ve } w_2 = (1 - \alpha)\alpha^{2-2} = (1-0,889) = 0,111 \text{ (} n = 2 \text{) olarak elde edilmiştir.}$$

$i + 1 \neq j$ olduğundan, (10) ve (12) denklemlerine göre $b = 0,889 \times 5 + 0,111 \times 3 = 4,778$ değerini alır. Böylece, birinci kriterin ikinci kritere göre bulanık değeri (1, 4,778, 7) şeklinde bulunmuştur.

Benzer şekilde birinci kriterin üçüncü kritere kıyasla bulanık değeri ÇY ve EDY arasında, böylece $i = 6$ ve $j = 9$ değerindedir. $\{a_L^i, a_M^i, a_M^{i+1}, \dots, a_M^j, a_R^j\} = \{1, 3, 5, 7, 9\}$ aralığının minimum ve maksimum değerleri olan a ve c parametreleri sırasıyla 1 ve 9 değerine sahiptir. b parametresini hesaplamak için;

$$\alpha = (g-j+i) / (g-1) = (10-9+6) / (10-1) = 0,778$$

$$w_1 = \alpha^{3-1} = 0,605, w_2 = (1 - \alpha)\alpha^{3-2} = (1-0,778) \times 0,778 = 0,173 \text{ ve } w_3 = (1 - \alpha) = 0,222 \text{ (} n = 3 \text{).}$$

$b = 0,605 \times 7 + 0,222 \times 5 + 0,173 \times 3 = 5,864$ değerine sahiptir. Böylece birinci kriterin üçüncü kritere göre bulanık değeri (1, 5,864, 9) olarak bulunmuştur.

Tablo 9: OWA Operatörü Aracılığıyla Elde Edilen Üçgen Bulanık Kümeler (Triangular Fuzzy Sets Found by OWA Operator)

| | f_1 | f_2 | f_3 | f_4 | f_5 |
|-------|-----------------------|----------------------|-----------------------|-----------------------|-----------------|
| f_1 | (1, 1, 1) | (1, 4,778, 7) | (1, 5,864, 9) | (0,14, 5,811, 9) | (1, 5,224, 9) |
| f_2 | (0,143, 0,209, 1) | (1, 1, 1) | (0,33, 4,9, 9) | (0,2, 4,928, 9) | (1, 5,864, 9) |
| f_3 | (0,111, 0,171, 1) | (0,111, 0,204, 3,03) | (1, 1, 1) | (0,11, 4,021, 7) | (0,2, 1,741, 5) |
| f_4 | (0,111, 0,172, 7,143) | (0,111, 0,203, 5) | (0,143, 0,249, 9,091) | (1, 1, 1) | (3, 5,222, 9) |
| f_5 | (0,111, 0,191, 1) | (0,111, 0,171, 1) | (0,2, 0,574, 5) | (0,111, 0,191, 0,333) | (1, 1, 1) |

OWA operatörü yardımıyla Tablo 9'da yer alan tüm üçgen bulanık kümeler hesaplandıktan sonra, Bölüm 2.2.2'de tarif edilen yöntemin 5. adımına göre her satır için geometrik ortalama değerler bulunmuştur. Örnek olarak, birinci satırın geometrik ortalaması şu şekilde hesaplanır:

$$a_g = (1 \times 1 \times 1 \times 0,14 \times 1)^{1/5} = 0,675$$

$$b_g = (1 \times 4,778 \times 5,864 \times 5,811 \times 5,224)^{1/5} = 3,854$$

$$c_g = (1 \times 7 \times 9 \times 9 \times 9)^{1/5} = 5,515$$

Böylece, birinci satırın geometrik ortalaması (0,675, 3,854, 5,515) olarak elde edilmiştir. Daha sonra, üçgen bulanık kümede yer alan tüm değerler Bölüm 2.2.2'de sunulan yöntemin 6. adımına göre normalize edilmiştir. Geometrik ortalamaları normalize etmek için Tablo 3'teki ölçeğin en yüksek değeri (9) referans alınmış, böylece $(f_1 \oplus f_2 \oplus \dots \oplus f_n)$ çarpanı olarak 9 değeri kullanılmıştır. Bu durumda, birinci maliyet kriterine ait üçgen bulanık kümenin normalize değerleri şu şekildedir:

$$a_w = 0,675 / 9 = 0,075$$

$$b_w = 3,854 / 9 = 0,428$$

$$c_w = 5,515 / 9 = 0,613$$

Tüm kriterlere benzer işlem uygulanarak elde edilen bulanık ağırlık değerleri Tablo 10'da verilmiştir.

Tablo 10: Maliyet Kriterlerinin Bulanık Ağırlık Değerleri (Fuzzy Importance Weights of Cost Criteria)

| Kriter | Bulanık Ağırlık Değerleri |
|--------|---------------------------|
| f_1 | (0,075, 0,428, 0,613) |
| f_2 | (0,044, 0,219, 0,415) |
| f_3 | (0,021, 0,084, 0,282) |
| f_4 | (0,039, 0,060, 0,548) |
| f_5 | (0,022, 0,036, 0,123) |

Her maliyet kriterinin nihai önem düzeyinin elde edilmesi için, Bölüm 2.2.2'de tarif edilen yöntemin 7. adımına göre Tablo 10'da yer alan değerlerin bulanıklığı kaldırılmış ve değerler normalize edilmiştir. Örnek olarak, birinci kriter $D_1 = (0,075 + 4 \times 0,428 + 0,613) / 6 = 0,400$ şeklinde hesaplanmıştır.

Tablo 11: Maliyet Kriterlerinin Bulanık Olmayan ve Normalize Ağırlık Değerleri (Defuzzified and Normalized Importance Weights of Cost Criteria)

| Kriter | Bulanık Olmayan Ağırlık Değerleri | Normalize (Net) Ağırlık Değerleri |
|--------|-----------------------------------|-----------------------------------|
| f_1 | 0,400 | 0,437 |
| f_2 | 0,223 | 0,243 |
| f_3 | 0,107 | 0,117 |
| f_4 | 0,138 | 0,151 |
| f_5 | 0,048 | 0,052 |

Tablo 11'de görüldüğü üzere, ara yüz ile ilgili nesne sayısını temsil eden kriter (f_1), %43,7 oranındaki ağırlık değeri ile yazılım projelerinin maliyetini etkileyen en önemli etmendir. Uygulama nesne sayısı (f_2), ikinci en önemli kriter olup %24,3 önem düzeyine sahiptir. Yığın iş ve veri tabanı nesne sayısı ile ilgili kriterler (f_3 ve f_4), %11,7 ve %15,1 değer ile birbirine yakın önemdedir. Son olarak, veri obje sayısı (f_5) ile ilgili kriterin yazılım

projelerinin maliyet tahmini açısından çok düşük öneme (%5,2) sahip olduğu görülmüştür.

Önerilen Klasik ve Sezgisel Bulanık İkili Karşılaştırma yöntemleri, Türkiye'de hizmet vermekte olan büyük bir yazılım şirketinde uygulanmıştır. Bu amaçla öncelikle, işletmede 2015 yılında tamamlanan 1.180 adet yazılım geliştirme faaliyeti için f_1, f_2, f_3, f_4 ve f_5 kriterleri ile temsil edilen nesne sayılarına ait gerçek bilgiler toplanmıştır. Daha sonra, Tablo 6 ve 11'deki ağırlık değerleri kullanılarak Klasik ve Sezgisel Bulanık İkili Karşılaştırma yoluyla ilgili çalışmaların maliyeti tahmin edilmiştir. Sonuç olarak, önerilen yöntemlerin etkinliğinin incelenmesi için gerçekleşen ve tahmini maliyet arasındaki sapma incelenmiştir. Tablo 12, Klasik ve Sezgisel Bulanık İkili Karşılaştırma yöntemleri ile tahmin edilen maliyetlerin gerçek maliyetlere uzaklığını göstermektedir.

Table 12: Önerilen Yöntemler ile Elde Edilen Sonuçların Gerçekleşen Maliyet ile Kıyaslanması (Benchmark of Estimated Cost Found by Proposed Methods with Real Effort)

| Gerçek Maliyet Aralığı (Adam-Gün) | Proje Sayısı | Ortalama Gerçek Maliyet (Adam-Gün) | Klasik İkili Karşılaştırma ile Ortalama Tahmini Maliyet (Adam-Gün) | Sezgisel Bulanık İkili Karşılaştırma ile Ortalama Tahmini Maliyet (Adam-Gün) | Klasik Yöntem Sapma (%) | Sezgisel Bulanık Yöntem Sapma (%) |
|-----------------------------------|--------------|------------------------------------|--|--|-------------------------|-----------------------------------|
| 0<x<10 | 164 | 6,74 | 5,98 | 6,06 | 11,28 | 10,09 |
| 10≤x<50 | 593 | 35,26 | 28,49 | 29,82 | 19,20 | 15,43 |
| 50≤x<100 | 358 | 64,08 | 74,85 | 70,95 | -16,81 | -10,72 |
| x ≥ 100 | 65 | 192,32 | 226,09 | 218,04 | -17,56 | -13,37 |

Tablo 12'de yer alan sapma değerleri, $1 - (\text{Tahmini} / \text{Gerçekleşen Maliyet})$ formülü ile hesaplanmıştır. Bu nedenle pozitif sapma değerleri, bahsedilen aralıktaki ortalama tahmini maliyetin gerçekleşen maliyetten daha düşük olduğu anlamına gelmektedir. Tablo 12'de görüldüğü üzere, Klasik İkili Karşılaştırma ile tahmin edilen maliyet; gerçek maliyeti 0-10 ve 10-50 adam-gün arasında olan talepler için sırasıyla ortalama %11,28 ve %19,20 daha düşük iken gerçek maliyeti 50-100 adam-gün arasındaki ve 100 adam-günden daha fazla olan talepler için sırasıyla ortalama %16,81 ve %17,56 daha yüksektir. Öte yandan, Sezgisel Bulanık İkili Karşılaştırma ile tahmin edilen maliyet; gerçek maliyeti 0-10 ve 10-50 adam-gün arasında olan talepler için sırasıyla ortalama %10,09 ve %15,43 daha düşük iken gerçek maliyeti 50-100 adam-gün arasındaki ve 100 adam-günden daha fazla olan talepler için sırasıyla ortalama %10,72 ve %13,37 daha yüksektir. Tablo 12'deki verilere göre, Sezgisel Bulanık İkili Karşılaştırma yönteminin Klasik İkili Karşılaştırma yönteminden daha başarılı sonuç verdiği görülmektedir. Özellikle Sezgisel Bulanık İkili Karşılaştırma ile proje bazında elde edilen tahmini maliyet değerleri, ikili karşılaştırma sırasında görüş bildiren uzmanlar tarafından doğrulanmıştır.

5. SONUÇLAR VE GELECEK ÇALIŞMALAR (CONCLUSION AND FUTURE RESEARCHS)

Bu çalışmada, yazılım projelerinin maliyetinin tahmin edilmesine yönelik yöntemler önerilmiştir. Bölüm 1’de tarif edildiği üzere; yazılım maliyetinin kestirilmesi amacıyla literatürde yaygın olarak kullanılan tekniklerden analitik olan ve olmayan yöntemlerin bazı eksiklikleri bulunmaktadır. Örneğin COCOMO vb. analitik yöntemlerde, uzman görüşünden faydalanmamakta ve sonucun doğruluğu büyük ölçüde kullanılan maliyet kriterleri ile fonksiyona bağlıdır. Öte yandan, uzmanların yeterli deneyime sahip olmaması ya da objektif olamaması durumunda, uzman görüşü vb. analitik olmayan yöntemler ile tahmin edilen maliyetler de gerçekten uzaklaşmaktadır. Ayrıca, konu ile ilgili literatürdeki çalışmalar incelendiğinde analitik ve subjektif yaklaşımları birleştiren çalışmalara rastlanamamıştır.

Bu çalışmada, analitik ve nesnel yöntemler bir arada kullanılarak her iki yaklaşımın avantajlarından yararlanılmıştır. Ayrıca literatürde, yazılım projelerinin maliyetinin genellikle kesin ya da meta sezgisel yöntemler ile tahmin edildiği görülmüştür. Özellikle belirsiz ortamlarda maliyet tahminleme için uygun yöntem olarak çalışmada önerilen Sezgisel Bulanık İkili Karşılaştırma tekniğinin uygulama örneğine literatürde rastlanmamıştır. Bu nedenle, uzman görüşü ile belirlenen en uygun kriterlerin göreceli önemi, yine uzman görüşünden yararlanarak Klasik ve Sezgisel Bulanık İkili Karşılaştırma ile tespit edilmiş, Sezgisel Bulanık İkili Karşılaştırma sonuçlarının gerçek maliyete daha yakın olduğu görülmüştür. Elde edilen sonuçlar, görüş bildiren uzmanlar tarafından doğrulanmıştır.

Gelecekte, bu çalışmada tespit edilen ağırlık değerleri kullanılarak yazılım maliyetinin minimize edilmesinin amaçlandığı matematiksel modeller oluşturulabilir; atanabilecek en fazla sayıda kaynak, en uzun tamamlama süresi vb. kısıtlar ile problemin en iyi çözümü bulunabilir. Ayrıca proje sayısının çok fazla olduğu durumlarda, problemin en iyi çözümünün bulunabilmesi için sezgisel ve meta sezgisel yöntemler geliştirilebilir.

KAYNAKLAR (REFERENCES)

- [1] S.W. Lin, V.M. Bier, “A Study of Expert Overconfidence”, *Reliability Engineering & System Safety*, 93, 711-721, 2008.
- [2] H. Arkes, “Overconfidence in Judgmental Forecasting” in J. S. Armstrong, *Principles of Forecasting*, Kluwer Academic Publishers, 495-515, 2001.
- [3] A. Stellman, J. Greene, “Applied Software Project Management”, O’Reilly Media, 2005.

- [4] K. Wiegers, “Stop Promising Miracles”, *Software Development*, 8, 49–53, 2000.
- [5] E. Miranda. “Improving Subjective Estimates Using Paired Comparisons”, *Software, IEEE* 87-91, 2001.
- [6] G.N. Parkinson. “Parkinson’s Law and Other Studies in Administration”. Boston, USA: Houghton-Mifflin, 1957.
- [7] R. Nelson. “Management Handbook for the Estimation of Computer Programming Costs”, *Systems Development Corporation*, 1966.
- [8] C.E., Walston, C.P. Felix. “A Method of Programming Measurement and Estimation. *IBM Systems Journal*, 16, 54-73, 1977.
- [9] B.W. Boehm. “Software Engineering Economics”, Englewood Cliffs, NJ, Prentice-Hall, 1981.
- [10] B.W. Boehm. “Software Cost Estimation with COCOMO II. New Jersey, Prentice Hall, 2000.
- [11] S. Bilgaiyan, S. Mishra, M. Das, “A review of software cost estimation in agile software development using soft computing techniques”, *2016 2nd International Conference on Computational Intelligence and Networks (CINE)*, Bhubaneswar, 2016, 112-117.
- [12] R. Sarno, J. Sidabutar, Sarwosri, “Comparison of different neural network architectures for software cost estimation”, *2015 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, Bandung, 2015, 68-73.
- [13] S.M.S. Jafari, F. Ziaaddini, “Optimization of software cost estimation using harmony search algorithm”, *2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)*, Bam, 2016, 131-135.
- [14] F.S. Gharehchopogh, R. Rezaii, B. Arasteh, “A new approach by using tabu search and genetic algorithms in software cost estimation”, *2015 9th International Conference on Application of Information and Communication Technologies (AICT)*, Rostov on Don, 2015, 113-117.
- [15] F.S. Gharehchopogh, I. Maleki, A. Talebi, “Using hybrid model of artificial bee colony and genetic algorithms in software cost estimation”, *2015 9th International Conference on Application of Information and Communication Technologies (AICT)*, Rostov on Don, 2015, 102-106.
- [16] T.R. Benala, S.C. Dehuri, S.C. Satapathy, C.S. Raghavi, “Genetic algorithm for optimizing neural

- network based software cost estimation”, *International Conference on Swarm, Evolutionary, and Memetic Computing*. Springer Berlin Heidelberg, 2011.
- [17] N. Kushwaha, Suryakant, “Software cost estimation using the improved fuzzy logic framework”, *2014 Conference on IT in Business, Industry and Government (CSIBIG)*, Indore, 2014, 1-5.
- [18] A. Mittal, K. Parkash, H. Mittal, “Software cost estimation using fuzzy logic”, *ACM SIGSOFT Software Engineering Notes* 35, 1-7, 2010.
- [19] C.L. Hwang, K. Yoon. “Multiple Attributes Decision Making Methods and Applications”, Berlin Heidelberg, Springer, 1981.
- [20] T.L. Saaty. “The Analytic Hierarchy Process”, McGraw – Hill Inc. 1980.
- [21] J.J Buckley, “Fuzzy Hierarchical Analysis”, *Fuzzy Sets and Systems*, 17, 233-247, 1985.
- [22] B. Zhu, Z. Xu, “Analytic Hierarchy Process-Hesitant Group Decision Making”, *European Journal of Operational Research*, 239, 794– 801, 2014.
- [23] B. Oztaysi, S.C. Onar, E. Boltürk, C. Kahraman, “Hesitant Fuzzy Analytic Hierarchy Process”, *FUZZ-IEEE* , 1-7, 2015.
- [24] T.L., Saaty. “How to Make a Decision: The Analytic Hierarchy Process”, *European Journal of Operational Research*, 48, 9–26, 1990.
- [25] L. A. Zadeh, “Fuzzy Sets”, *Information and Control*, 8(3), 338–353, 1965.
- [26] F.T. Bozbura, A. Beskese, C. Kahraman, “Prioritization of Human Capital Measurement Indicators Using Fuzzy AHP”, *Expert Systems with Applications* 32, 1100-1112, 2007.
- [27] V. Torra, “Hesitant Fuzzy Sets”. *International Journal of Intelligent Systems*, 25, 529–539, 2010.
- [28] H. Liu, R.M. Rodriguez, “A Fuzzy Envelope for Hesitant Fuzzy Linguistic Term Set and Its Application to Multi-Criteria Decision Making”, *Information Sciences*, 258, 220–238, 2014.
- [29] D. Filev, R.R. Yager, “On the Issue of Obtaining OWA Operator Weights”, *Fuzzy Sets and Systems*, 94, 157-169, 1998.
- [30] I. Jacobson, “Object-Oriented Software Engineering: A Use Case Driven Approach”, **Addison-Wesley**, 1993.
- [31] K.. Ribu, “**Estimating Object-Oriented Software Projects with Use Cases**. MSc Thesis, Oslo, Norway, 2001.
- [32] H.M. Sneed, “Estimation of the Development Costs of Object-Oriented Software”, *Informatik Spektrum*, 19, 133-140, 1996.
- [33] T.W. Koh, M.H. Selamat, A.A.A. Ghani, R. Abdullah, “Review of Complexity Metrics for Object-Oriented Software Products”, *International Journal of Computer Science and Network Security*, 8, 314-320, 2008.