

**VEKİL MODELLER KULLANILARAK ETKİN TOLERANS ANALİZİ**

**UTKU BAYRAM**

**YÜKSEK LİSANS TEZİ**

**MAKİNE MÜHENDİSLİĞİ**

**TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ**

**FEN BİLİMLERİ ENSTİTÜSÜ**

**ARALIK 2014**

**ANKARA**

Fen Bilimleri Enstitü onayı

---

Prof. Dr. Osman EROĞUL  
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

---

Doç. Dr. Murat Kadri AKTAŞ  
Anabilim Dalı Başkanı

Utku BAYRAM tarafından hazırlanan VEKİL MODELLER KULLANILARAK ETKİN TOLERANS ANALİZİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

---

Doç. Dr. Erdem ACAR  
Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Gültekin KUYZU

Üye : Doç. Dr. Erdem ACAR

Üye : Yük Müh. Burak Durak

## **TEZ BİLDİRİMİ**

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

**Utku BAYRAM**

<b>Üniversitesi</b>	: TOBB Ekonomi ve Teknoloji Üniversitesi
<b>Enstitüsü</b>	: Fen Bilimleri
<b>Anabilim Dalı</b>	: Makine Mühendisliği
<b>Tez Danışmanı</b>	: Doç. Dr. Erdem ACAR
<b>Tez Türü ve Tarihi</b>	: Yüksek Lisans – Aralık 2014

## **Utku BAYRAM**

### **VEKİL MODELLER KULLANILARAK ETKİN TOLERANS ANALİZİ**

#### **ÖZET**

Günümüzdeki rekabetçi ortamda düşük maliyetli ve kaliteli ürünler sunmak en önemli hedeflerden biri haline gelmiştir. Kaliteyi düşürmeden üretim maliyetlerini aşağı çekmek birçok disiplinin bir arada çalışmasıyla mümkün olur. Üretim toleransları kontrol edildiği sürece çok etkin bir şekilde maliyetleri hızla düşürmek mümkündür. Günden güne artan bilgisayar gücüyle birlikte modelin artan karmaşıklığı ve tolerans detayları, tolerans analizini zorlaştırmaktadır. Son yıllarda zahmetli geleneksel analiz kodları/yöntemleri yerine, daha hızlı ve aynı zamanda güvenilir model veya benzetimler kullanılmaya başlanmıştır. Çeşitli alanlarda sıkça kullanılan vekil modeller, düşük hesapsal maliyetleri ve güvenilir sonuçları nedeniyle, zahmetli ve hesapsal maliyeti yüksek geleneksel çözüm yöntemleri yerine kullanılmaktadır. Bu çalışmada, vekil modellerin tolerans analizinde etkin bir şekilde uygulanabilir olduğunu göstermek amaçlanmıştır. Geleneksel tolerans analiz yöntemleri olan WC, RSS ve MCS ile bu çalışmada önerilen yanıt yüzey ve Kriging vekil modelleri tabanlı tolerans analizi yöntemleri, doğrusal olmayan iki boyutlu bir örnek problem ve üç boyutlu daha karmaşık bir örnek problem üzerinde uygulanmıştır. Vekil model tabanlı tolerans analizi yöntemlerinin geleneksel tolerans analizi yöntemlerinden daha etkin olduğu gözlenmiştir. Ayrıca, Kriging modeli ile yapılan tolerans analizinin yanıt yüzey modeli ile yapılan tolerans analizinden daha doğru sonuçlar verdiği gözlenmiştir.

**Anahtar Kelimeler:** tolerans analizi, vekil model, yanıt yüzey, Kriging, WC, RSS, Monte Carlo

**University** : TOBB Economics and Technology University  
**Institute** : Institute of Natural and Applied Sciences  
**Science Programme** : Mechanical Engineering  
**Supervisor** : Assoc. Prof. Erdem ACAR  
**Degree Awarded and Date** : M.Sc. – December 2014

**Utku BAYRAM**

**EFFICIENT TOLERANCE ANALYSIS BY USING SURROGATE MODELS**

### **ABSTRACT**

Producing low cost products with high quality has become one of the most important goals of the production in today's competitive world. Decreasing production costs without increasing quality loss is possible through working in a multi disciplinary manner. As long as the manufacturing tolerances are controlled, the production costs can efficiently be reduced. Continuing growth of computing power and speed increases the complexity of assembly models and tolerance details, thereby the tolerance analysis becomes difficult. In the last decades, inexpensive yet reliable models or simulation techniques have become available in place of overwhelming traditional computer analysis codes/techniques. Surrogate models, which are widely used in many research fields, are usually used in place of overwhelming and computationally expensive traditional methods, due to their low computational cost and reliable solutions. In this study, the main objective is to show that surrogate models are suitable for tolerance analysis. Traditional tolerance analysis methods such as WC, RSS and MCS along with response surface model and Kriging surrogate model based tolerance analysis methods are applied on a two dimensional nonlinear example problem and three dimensional example problem which is more complicated than the two-dimensional problem. It is observed that the surrogate based tolerance analysis methods are more efficient than the traditional tolerance analysis methods. It is also observed that Kriging based tolerance analysis is more accurate than response surface based tolerance analysis.

**Keywords:** tolerance analysis, surrogate models, response surface, Kriging, WC, RSS, Monte Carlo

## **TEŐEKKÜR**

Yüksek lisans eğitiminin boyunca desteęini ve ilgisini eksik etmeyen, anlayışı, sabrı ve yol göstericilięi ile bilgi birikiminden yararlandığım, örnek aldığım tez danışmanım olan değerli hocam Doç Dr. Erdem ACAR'a teşekkür ederim.

## İÇİNDEKİLER

TEZ BİLDİRİMİ.....	iii
ÖZET.....	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	xi
KISALTMALAR.....	xii
SEMBOLLER.....	xii
1. GİRİŞ.....	1
1.1. Tolerans.....	1
1.2. Tolerans Tasarımı.....	4
1.3. Literatür Taraması.....	4
1.4. Amaç ve Kapsam.....	6
2. YÖNTEMLER.....	7
2.2. Tolerans Analizi Yöntemleri.....	7
2.3. En Kötü Durum Analizi.....	7
2.4. İstatistiksel Yöntemler.....	8
2.4.2. Monte Carlo Simülasyonu.....	11
2.5. Vekil Model Yöntemleri.....	13
2.5.1. Yanıt Yüzey Yöntemi.....	14
2.5.2. Kriging Yöntemi.....	16
2.6. Bilgisayar Destekli Yazılımlarla Tolerans Analizi.....	18

2.6.1. TolAnalyst.....	20
3. BİR KAVRAMA ELEMANININ TOLERANS ANALİZİ (2 BOYUTLU PROBLEM) .....	22
3.2. Problemin Tanımı.....	22
3.3. Geleneksel Çözüm.....	23
3.4. Vekil Model Aracılığıyla Çözüm .....	25
3.5. Bulgular .....	26
4. BİR DİŞLİ KUTUSUNUN TOLERANS ANALİZİ (3 BOYUTLU PROBLEM) 30	
4.2. Problemin Tanımı.....	30
4.3. TolAnalyst Yardımıyla Çözüm .....	34
4.4. Vekil Model Aracılığıyla Çözüm .....	41
4.5. Bulgular .....	42
5. DEĞERLENDİRME.....	45
5.2. Sonuçlar ve Tartışma.....	45
5.3. İleriye Yönelik Çalışmalar .....	47
6. EKLER.....	48
KAYNAKLAR .....	74
ÖZGEÇMİŞ .....	77



## ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 1-1 Geometrik karakteristik sembolleri [4].....	3
Çizelge 2-1 Standart sapma sayılarına göre kalite karşılaştırması.....	9
Çizelge 3-1 Rassal değişkenlerin ortalama ve standart sapmaları .....	23
Çizelge 3-2 Fortini'nin kavrama montajı için MCS sonuçları .....	27
Çizelge 3-3 Fortini'nin kavrama montajına uygulanan 40 örneklemler RSM sonuçları .....	28
Çizelge 3-4 Fortini'nin kavrama montajına uygulanan 40 örneklemler KM sonuçları.....	29
Çizelge 3-5 TolAnalyst ile elde edilen sonuçlar .....	42
Çizelge 3-6 Dişli Kutusu Montajı için MCS sonuçları .....	43
Çizelge 3-7 Dişli Kutusu Montajı için RSM ile elde edilen sonuçlar.....	43
Çizelge 3-8 Dişli Kutusu Montajı için KM ile elde edilen sonuçlar.....	44

## EKLERİN LİSTESİ

<b>EK</b>	<b>Sayfa</b>
EK 1 rss_mcs_clutch.m fonksiyon kodları .....	48
EK 2 mcs_clutch.m fonksiyon kodları.....	49
EK 3 mcs_vekil_clutch.m fonksiyon kodları.....	50
EK 4 rsm_clutch.m fonksiyon kodları .....	51
EK 5 statistic.m fonksiyon dosyası .....	53
EK 6 krig_clutch.m fonksiyon dosyası .....	54
EK 7 Fortini'nin Kavrama Montajı için RSM sonuçları.....	56
EK 8 Fortini'nin Kavrama Montajı için KM sonuçları.....	57
EK 9 gbox_rand.m fonksiyon dosyası .....	58
EK 10 gbox_macro.swp macro dosyası ve modulleri.....	61
EK 11 gbox_rsm fonksiyon dosyası .....	70
EK 12 gbox_rand.m fonksiyon dosyası .....	72

## ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 1-1 Tolerans analizi [9].....	4
Şekil 1-2 Tolerans sentezi [9] .....	4
Şekil 2-1 Süreç kapasitesi $3\sigma$ olan rastgele hatalı sürecin normal dağılım grafiği ....	10
Şekil 2-2 MCS ile montaj tolerans analizi .....	12
Şekil 3-1 Fortini'nin Kavrama Montajı .....	22
Şekil 3-2 Fortini'nin Kavrama Montajı için 100,000 örneklemlile MCS histogramı ...	28
Şekil 3-3 Dişli Kutusu Montaj elemanlarının görüldüğü kesit .....	31
Şekil 3-4 Muhafaza elemanının geometrik toleransları .....	32
Şekil 3-5 Yuvarlak Kapak Plakasının geometrik toleransları .....	33
Şekil 3-6 Sonsuz Dişli Şaftın geometrik toleransları .....	33
Şekil 3-7 Kritik ölçü için unsurların sıralı seçimi .....	35
Şekil 3-8 Yuvarlak Kapak Plakası-1 için montaj sınırlandırmaları sıralı gösterimi ..	36
Şekil 3-9 Sonsuz Dişli Şaft-1 için montaj sınırlandırmaları sıralı gösterimi .....	36
Şekil 3-10 Yuvarlak Kapak Plakası-2 için montaj sınırlandırmaları sıralı gösterimi	37
Şekil 3-11 Dişli Kutusu Montajına katkı sağlayan eleman ve toleransların listesi ile en yüksek duruma göre yüzde katkıları .....	38
Şekil 3-12 Pozisyon tolerans değişimi için uygulanan yöntemin gösterimi .....	40

## KISALTMALAR

<b>Kisaltmalar</b>	<b>Açıklama</b>
<b>CMM</b>	Coordinate Measuring Machine
<b>WC</b>	Worst Case
<b>RSS</b>	Root Sum Square
<b>DOE</b>	Design of Experiments
<b>API</b>	Application Programming Interface
<b>CAT</b>	Computer Aided Tolerancing
<b>DACE</b>	Design Analysis of Computer Experiments

## SEMBOLLER

<b>Simgeler</b>	<b>Açıklama</b>
$\sigma$	Standart sapma
$a$	Tolerans hassasiyeti
T	Tolerans
<b>İndisler</b>	<b>Açıklama</b>
i	Montaj bileşeninin sayısı
n	Montajdaki bileşen adedi
asm	Montaj
<b>Üsler</b>	<b>Açıklama</b>
T	Transpozu
'	Tersi

## 1. GİRİŞ

1920'lerin sonlarında sanayileşmeden önce, ürünlerin montajındaki en önemli değerlendirme, parçaların planlandığı şekilde birbirine geçmesi ve kısa vadede günlük ihtiyaçları karşılayacak şekilde fonksiyonel olmasından ibaretti. Sınırlı üretim kapasitesine sahip olan imalathanelerde ürünün çalışıyor olma yeterliliği, 1970'lerin başlarında sanayinin gelişmesi ve seri üretime geçilmesiyle birlikte ortadan kalktı ve başka sorunların da var olduğu ortaya çıktı. Çünkü yüksek miktarda üretilen ürünler her zaman planlandığı şekliyle üretilmediği gibi ürünlerin standartı da yoktu.

Geçmişte teknik resimler üzerinde verilen ölçülerin sapması, doğrusal sınırlandırılmalar ile ifade edilerek kontrol edilmeye çalışılmıştır. Fakat ideal ölçülerle ifade edilen boyutların gerçekte üretilmemesi birçok nedene bağlıdır; örneğin takım aşınması, süreç farkı, kurulum hataları, malzeme özellikleri, sıcaklık, el işçiliği farkı gibi. Günümüzde dahi tasarımdan ürünün son haline kadar her türlü etki kontrol edilmeye çalışılmaktadır [1]. Tasarımcı tarafından fonksiyonellik, ömür, değiştirilebilirlik gibi birçok konu göz önünde bulundurularak sınırlandırılır. Bu sınırlı varyasyon ise tolerans olarak kabul edilir ve ideali sıfıra en yakın olanıdır.

### 1.1. Tolerans

Tolerans, "işlenmiş bir parçanın yapım ölçüsünde olabilecek özür payı" olarak tanımlanabilir [2]. Tolerans tasarım ve imalatın en önemli unsurudur. Tolerans aynı zamanda üretim döngüsündeki birçok aşamada ilgili bölümlerin ürün üzerinde iletişim kurmasını sağlar. İlkel dönemlerde tasarımcı kendi tecrübelerine ya da fonksiyonellik ölçüsüne dayanarak diğer bölümleri zor durumda bırakabiliyordu. Ya da imalat bölümü kapasitelerine göre tasarımı kısıtlayabiliyordu. Zamanla birlikte ortaya çıkmıştır ki, tasarım ve imalat faaliyetlerinin bir denge içinde yürütülmesi maliyetleri büyük oranda düşürmektedir. Örneğin; fonksiyonelliği garanti altına almak için dar toleranslar verilebilir, fakat kabiliyetlerin yetersizliği nedeniyle bu toleranslara sahip ürünler üretilemezse ortaya çıkan hurda ve hatalı ürün sayısı

maliyeti arttırır. Geniş verilen toleranslar ise imalat maliyetlerini düşürmesine rağmen, ürünün fonksiyonelliğini uzun vadede yitirmesine ve müşteri memnuniyetsizliği ile tekrar maliyetlerin artmasına neden olur.

Aslında toleranslar üretim döngüsünün her aşamasında farklı amaçlarla kullanılır ve her bir aşama birbirine bağlıdır. Örneğin; imalat bölümü tasarım bölümünden gelen toleranslara göre üretim yaparken, tasarımcı imalat kapasitesine göre tasarımını günceller. Montaj bölümü fiziksel olarak montajlanabilme gereksinimi ile test bölümünden gelen fonksiyonellik gereksinimleri karşılayamadığında, tasarım bölümü önerisini günceller. Kalite bölümü ise imalat çıktılarını tasarım bölümünden gelen tolerans gereksinimine göre kontrol ederken, test ve montaj bölümüyle bu değerler doğrulanır. Tüm bu bilgiler fiziksel (parça/montaj) ya da bilgi akışıyla döndürülür. Tüm bu olaylar ve organizasyonu düzenlemek amacıyla eş zamanlı mühendislik kavramı ortaya çıkmıştır [3].

Tolerans konusu tek bir açıdan incelenecek bir konu olmayıp literatürde yapılan çalışmalar şu şekilde gruplandırılabilir; toleranslama şemaları, tolerans modellemesi ve sunumu, tolerans spesifikasyonları, tolerans analizi, tolerans sentezi, tolerans taşınması, tolerans değerlendirmeleri. Tolerans şeması parametrik ve geometrik olarak iki kısma ayrılır. Parametrik olan belirli kategorilere sınıflandırılmış tolerans kartlarını ifade ederken, en basit olanı günümüzde halen kullanılan +/- boyutsal toleranslardır. Diğeri ise GD&T olarak bilinen geometrik toleranslardır. GD&T karakteristik olarak parçanın formunu, yerleşimi, konumu ve profil özelliklerini kontrol eder. Çizelge 1-1'de, kontrol edilen geometrik toleranslar ve sembolleri gösterilmiştir [4]. Bazı kuruluşlar tarafından bu toleransların tipi ve değerleri bakımından nasıl ifade edileceği spesifikasyon olarak yayınlanmıştır. Bu kuruluşlardan en önemli ikisi ASME ve ISO'dur. Bu kuruluşların tolerans spesifikasyonları ASME Y14.5 [4] ve ISO 1101 [5] olarak bilinir ve bu iki standart dünya üzerinde kullanılanların %90'ını kapsar ve birbirleri arasında %70'in üzerinde bir uyum vardır [6]. Bunun dışında ANSI, DIN veya firmanın kendi spesifikasyonları da mevcuttur [6].

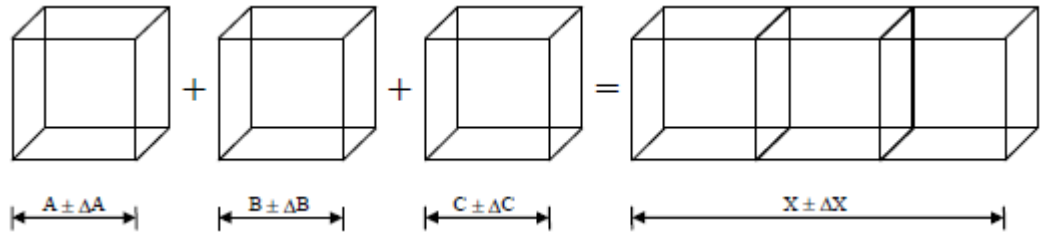
Çizelge 1-1 Geometrik karakteristik sembolleri [4]

UYGULAMA	TOLERANS TİPİ	KARAKTERİSTİĞİ	SEMBOL
BİREYSEL ÖZELLİKLER	FORM	DOĞRUSALLIK	—
		YÜZEY DÜZGÜNLÜĞÜ	
		DAİRESELLİK	
		SİLİNDİRİKLİK	
BİREYSEL YA DA İLİŞKİLİ ÖZELLİKLER	PROFİL	ÇİZGİ PROFİLİ	
		YÜZEY PROFİLİ	
İLİŞKİLİ ÖZELLİKLER	YERLEŞİM	AÇISALLIK	
		DİKLİK	
		PARALELLİK	
	KONUM	POZİSYON	
		EŞ EKSENLİLİK	
		SİMETRİ	
	YALPA	RADYAL SALGI	
		TOPLAM SALGI	

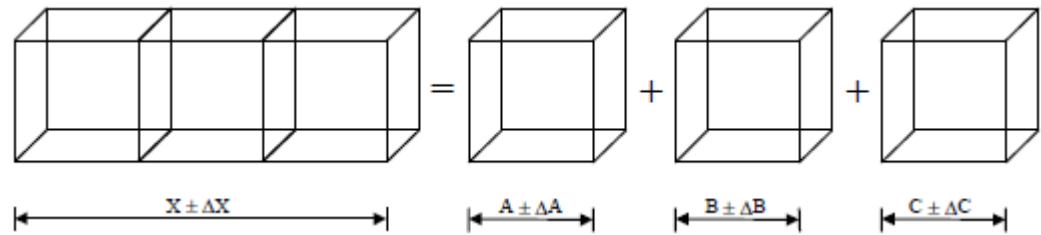
Bir diğer araştırma konusu, kendi çalışması içinde resmi olmayan haliyle matematiksel ya da elektronik olarak ifade edilmesi için yapılan araştırmaların olduğu bir alan olan, tolerans modelleme ve sunumudur. Tolerans transfer ise tasarım ve imalat arasında bir köprü kurarak, makinedeki kapasite ve hassasiyeti dikkate alan ya da takım aşınması ve makine hataları dikkate alarak tasarımcı için kolaylık sağlayan çalışma alanıdır. Diğer yandan tolerans dağıtımı da bu alanda sayılabilir. Tolerans değerlendirmesi ise kalite bölümüyle ilgili olup verilen toleransların koordinatlı ölçüm cihazları (CMM) aracılığı ile nasıl daha verimli ölçüleceğinin araştırıldığı alandır [7].

## 1.2. Tolerans Tasarımı

Yukarıda bahsedilen çalışma alanlarının dışında en çok ilgi duyulan alan tolerans tasarımı konusudur. Bu konu tolerans analizi ve tolerans sentezi (tolerans dağıtımı olarak da bilinir) bölümlerini içerir. Tolerans analizi Şekil 1-1’de gösterildiği gibi her bileşenin toleransını ve özelliklerini kullanarak montajda bulunan kritik boşluğun ya da ölçünün toleransını bulmaya çalışır. Tolerans sentezi ise Şekil 1-2’de gösterildiği gibi analizin tam tersi olarak, toleransı bilinen montajın bileşenlerinin özelliklerine uygun olarak bileşenler arasında dağıtılmasıdır. Tolerans analizi daha karmaşık ve ürüne bağlıken, tolerans sentezi daha çok imalat ve maliyet konuları üzerinden en iyileme yapılan konuları içerir [8].



Şekil 1-1 Tolerans analizi [9]



Şekil 1-2 Tolerans sentezi [9]

## 1.3. Literatür Taraması

Tolerans analizi yöntemleri incelendiğinde son kırk yıldır bu alanda aktif araştırma yapıldığı görülmektedir. Tolerans analizi konusu, temel yaklaşım ve gelişmiş yaklaşım olarak gruplandırılabilir. Klasik yöntemler olarak da ifade edilebilen temel



yaklaşım yöntemleri, ilk olarak 1980'lerin sonunda Fortini'nin kitabında [11] özetlemiştir. Bu yöntemler en kötü durum (WC) ve istatistiksel (kareler toplamının karekökü veya RSS) yöntemleridir. Yapılan ilk çalışmalarda tek boyutlu doğrusal problemler üzerinde çalışılmıştır, ardından Greenwood ve Chase [8] ile Mansoor [12] tarafından yapılan çalışmalarda ise basit bir gözlemlerle süreçlerdeki değişiklikleri ya da kaymaları da dikkate alarak daha gerçekçi bir sonuç elde etmek için yaklaşık (veya tahmini) WC ve RSS formülleri sunulmuştur.

Gelişmiş tolerans analizi yaklaşımları üzerine yapılan çalışmalar ise, süreç dağılımının değişkenliğine göre ve montaj yanıt fonksiyonunun karmaşıklığına bakılarak ikiye ayrılabilir. Chase ve ekibi [14] 1990'ların sonuna doğru direkt doğrusallaştırma yöntemi ile klasik analiz yöntemini kullanarak, daha karmaşık ve 2 ve 3 boyutlu mekanik montajlar için uygulamıştır. Benzer şekilde Zhang ve Wang [13] kam mekanizması kullanarak Chase ve ekibinin kullandığı direk doğrusallaştırma yöntemini uygulamıştır ve bununla birlikte tolerans sentezi de yapmıştır. Gao ve ekibi [15] direkt doğrusallaştırılmış modeli Monte Carlo yöntemi ile çözerek uygulamıştır. Karmaşık montajlarda görülen tasarım fonksiyonlarının daha kolay çözülebilir olması için Taylor seri açılımını kullanan Greenwood ve ekibi [16] birkaç yıl sonra, aynı yöntemi RSS için de denemiştir. Tüm bunlarla birlikte karmaşık tasarım fonksiyonlarını ile ifade edilen problemler için Hasofer-Lind güvenilirlik indisi yönteminin de kullanıldığı görülmektedir. Performans olarak doğrusal olmayan sistemlere bakılacak olursa, göreceli işlem gücü oranının WC için 1 birim, Hasofer-Lind için 6 birim, Method of Moments için 10 birim, Monte Carlo için 100 birim olduğu düşünülebilir [8].

Tolerans analizi çalışmaları süreç dağılımların karmaşıklığına göre incelenecek olursa, RSS yönteminde varsayılan Gaussian dağılımı (yani normal dağılım) her zaman doğru kabul edilemez. Bunu gören Lin ve ekibi [17] normal dağılım yerine tolerans analizinde beta dağılımını incelemiştir. Bunun dışında Gladman tarafından dinamik ortalama kayması, Desmond tarafından olasılıksal ortalama kayma modeli ve bunların bir arada kullanımı ortaya atılmıştır [18]. Genel olarak bu çalışmalar ilerleyen yıllarda analiz açısından çok sık kullanılmış olmayıp, tolerans sentezinde

maliyet hesaplarının en iyilemesinde (örneğin; geciken toptan ürün tedarikinin neden olduğu farklılıklar, zamanla oluşan hatalardaki değişim, seri üretimdeki ürün farklılıkları gibi) kullanılmıştır [19].

Genel bir değerlendirme yapılacak olursa; tolerans zincir modelinin birçok yönüyle incelenildiği, iki ve üç boyutlu modellerde uygulanabilir olduğu, fakat problem karmaşık bir hal aldığına bir hayli zor ve zahmetli olduğu, dolayısıyla bilgisayar destekli otomatikleştirilebilir çalışmalarının araştırıldığı görülmektedir. Benzetim tabanlı yöntemlerden MCS nin en çok tercih edilen yöntem olduğu, fakat tolerans sentezinde bazı fonksiyonlara uygulanamadığı da görülmüştür.

#### **1.4. Amaç ve Kapsam**

Yukarıda incelenen çalışmalar doğrultusunda en büyük problem, geleneksel yöntemler ile uygulanan tolerans analizinin her durumda ve her türlü geometrik tolerans için uygulanabilirliğinin güç olmasıdır. Popüler bir benzetim tekniği olan Monte Carlo yöntemi ise, kolay uygulanabilir olmasına rağmen hesapsal maliyeti en yüksek olanıdır. Burada doğrusal olmayan karmaşık problemlerden tasarım fonksiyonunu oluşturmak, 2 ve 3 boyutlu problemlerden tolerans zinciri oluşturulsa bile bileşenlerin hassasiyetini hesaplamak çok güç ve maliyeti yüksek görünmektedir.

Bu çalışmada, yukarıda bahsedilen zorlukların üstesinden gelebilmek için, daha genel bir bakış açısıyla vekil model tabanlı tolerans analizi konusuna girilecektir. İki ve üç boyutlu problemler üzerinde, mevcut klasik yöntemler olan WC, RSS ve MCS ile güvenilir sonuçlar elde edilecek ve vekil model tabanlı tolerans analizi yaklaşımı ile elde edilen sonuçlarla karşılaştırılacaktır. Bu çalışmada, yanıt yüzey ve Kriging vekil modelleri kullanılacak ve birbirleriyle mukayeseleri de yapılacaktır.

## 2. YÖNTEMLER

### 2.2. Tolerans Analizi Yöntemleri

Literatürde örneklenen şekliyle genel olarak kabul görmüş ve birçok açıdan incelenmiş tolerans analizi (yığılması veya birikimi) yöntemleri bu bölümde incelenecektir. Geleneksel yöntemler olarak da adlandırılabilir olan WC, RSS ve MCS yöntemleri anlatılacaktır.

### 2.3. En Kötü Durum Analizi

En bilinen tolerans analizi modeli olan en kötü durum analizi (WC), montaj bileşenlerinin toleranslarının doğrusal olarak mutlak değerlerinin toplanmasıyla hesaplanır. Böylece kritik bölgenin toleransı, fazla ya da eksik üretilmiş bileşenlerin en kötü durumdaki birleşimiyle elde edilir. Yöntem, aşağıdaki şekilde matematiksel olarak ifade edilebilir.

$$T_{asm} = \sum_{i=1}^n |a_i T_i| \quad (2.1)$$

Burada  $T_{asm}$  montajın eşit iki taraflı toleransı  $T_i$  ise her bir  $i$  nci bileşenin eşit iki taraflı toleransı olarak kabul edilir.  $a_i$  ise burada hassasiyetini, yani toleransın montaja etkisini ifade eder ve tek boyutlu hesaplamalarda  $\pm 1$  dir, bazı tolerans ifadeleri gereği 0,5 olarak da tek boyutlu hesaplamalarda kullanılabilir. Tolerans yığılmasındaki her bir tolerans, birbirinden bağımsız birer değişkendir ve sayısı  $n$  olarak ifade edilir.

Bu yöntemin doğası gereği her bir montaj bileşeninin ölçüsü, en fazla ya da en düşük sınırında üretildiği varsayılır. Dolayısıyla böyle bir durumun gerçekleşmesi için üretilen bütün parçaların %100 kalite kontrolünün yapıldığı varsayılır. Dolayısıyla

kontrol edilen bu parçaların hiç birinin verilen toleranslar dışında olmadığı kabul edilir.

Tüm bu varsayımlar nedeniyle bazı dezavantajlar doğar. Yapılan varsayımlar çoğu kez gerçekçi olmadığı için hatalı olması muhtemeldir. Uygulanan bu yöntem genellikle incelenilen kritik ölçünün çok dar bir tolerans aralığında olması gerektiğini ortaya koyar. Bu da uygulamada üretim maliyetinin artmasına neden olur. Bu yöntem genelde çok kritik montaj parçalarında ya da çok az bileşeni olan montajlarda kullanılır. Diğer bir yandan bu hesaplama yöntemi bazen en düşük boşluk değeri 0 olması gibi gereksinimleri karşılamayabilir. Bu gibi durumlarda boyutlandırma faktörü hesaplanarak tüm toleranslar belli oranda boyutlandırılır [20].

#### **2.4. İstatistiksel Yöntemler**

İstatistiksel yöntemler, tolerans ifadeleriyle birlikte sıkça kullanılan gerçekçi bir yaklaşımdır. İstatistiksel tolerans hesaplamalarında genelde montaj ve bileşen toleranslarının normal dağılımlı olduğu ve her bir toleransın birbirinden istatistiksel olarak bağımsız oldukları varsayılır. Yöntem esnek olup Beta, Poisson gibi dağılımları da (eğer gerekirse) çözüme uygulamak mümkündür.

WC yöntemine kıyasla istatistiksel yöntemler daha esnektir ve her bir toleransın dağılımı, birbirini etkileyen tolerans birikimi içindeki değerler ile denge oluşturur. Yani bir bileşenin toleransı biraz fazla iken diğerinin daha fazla olmasına izin verir. Böylece hatalar dengelenir ve toplamsal sonuçta daha geniş bir tolerans aralığı elde edilmesini sağlar. Bu da imalat maliyetlerini düşürür. Bu yöntemde, belli bir yüzde ile parçanın toleransının tasarım sınırlanmaları dışında üretildiği de göz önünde bulundurulur. Böylece daha gerçekçi ve geniş bir çözüm elde edilmiş olunur. Aslında bu model parça üretilirken malzeme, sıcaklık, takım aşınması, işçilik hataları gibi varyasyonların olasılıksal olarak çıktısını da gerçekçi bir biçimde göz önüne buldurmuş olur. Böylece verimlilik ve tasarım sınırları dışına çıkan hurda değerlerinin de maliyet açısından hesaplanabilmesine imkân sağlar.

İstatistiksel yöntemler kullanılırken genel olarak ortalama ( $\mu$ ) olasılık dağılımı eğiminin en üst noktasını, popülasyonun standart sapması ise ( $\sigma$ ) üretim hassasiyeti ve süreç kapasitesini ifade eder. Taguchi'nin Altı Sigma kalite seviyelerine göre karşılaştırılırsa, Çizelge 2-1'deki gibi tasarım sınırları dışında kalan bileşenlerin sayısı elde edilebilir [21].

Çizelge 2-1 Standart sapma sayılarına göre kalite karşılaştırması

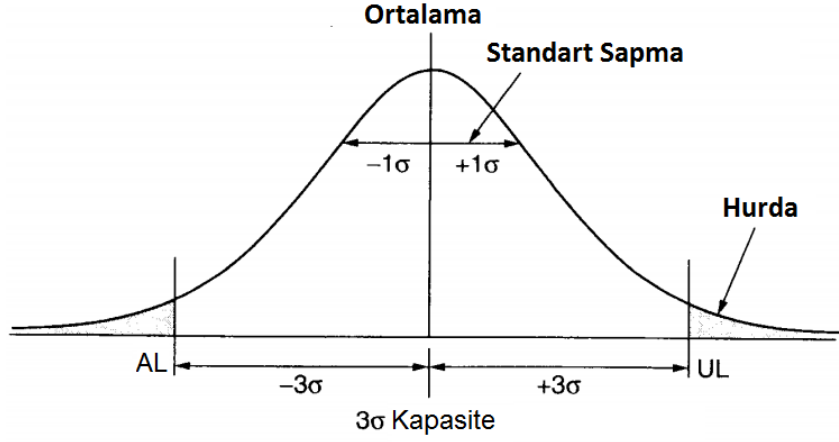
<b>Sigma</b>	<b>Milyonda Hurda</b>	<b>Milyonda hurda</b>	<b>Kalite</b>
$\pm 1\sigma$	690,000	Üretimin >%40	Kabul edilemez
$\pm 2\sigma$	308,537	Üretimin %30-40	Kabul edilemez
$\pm 3\sigma$	66,807	Üretimin %20-30	Makul
$\pm 4\sigma$	6210	Üretimin %15-20	Yüksek
$\pm 5\sigma$	233	Üretimin %10-15	Çok yüksek
$\pm 6\sigma$	3.4	Üretimin <%10	Aşırı yüksek

#### 2.4.1. Kareler Toplamının Karekökü

Kareler Toplamının Karekökü (RSS) yöntemi, ortalama kabul edilen verilmiş nominal ölçü ve tolerans sınırlarını normal dağılım içerisinde kapsayacak varsayımıyla çözer. Bu nedenle verilen boyutsal tolerans  $\mu \pm n\sigma$  şeklinde ifade edebilir. Burada  $n$  süreç kapasitesini ifade eder. Örneğin 3 sigma süreç kapasitesi için bir bileşenin ölçüsünün histogramı Şekil 2-1'deki gibi gösterilebilir. Şekil 2-1'de  $AL$  alt limit,  $ÜL$  ise üst limiti gösterir. Tasarım gereksinimlerinde verilen tolerans sınırlarıdır.

RSS denklem 2.2 de verildiği gibi hesaplanır.

$$T_{asm} = \sqrt{\sum_{i=1}^n a_i^2 \cdot T_i^2} \quad (2.2)$$



Şekil 2-1 Süreç kapasitesi  $3\sigma$  olan rastgele hatalı sürecin normal dağılım grafiği

Denklem (2.1)'de her bir  $i$ 'nci bileşenin montaja olan etkisi olarak ifade edilir. Eğer montaj fonksiyonu  $f(x_i)$  ise 2 ya da 3 boyutlu doğrusal olmayan fonksiyonlarda  $a_i = \partial f / \partial x_i$  şeklinde düşünülebilir. Bu  $i$ 'nci bileşenin nominal ölçüsü  $x$ 'in parçalı türevini yani montajdaki hassasiyetini ifade eder [18].

Bu çözüm dağılımın ortalamasının sabit ve AL ile ÜL arasında orta noktada olduğunu varsayar. Fakat gerçekte çoğu zaman bu mümkün değildir. Tüm süreçler ve çıktıları zaman içinde kayar. Bunların nedeni takım aşınması, sıcaklık etkileri, elektronik sistemlerin sistematik hataları ve işçilik olabilir. Bunlar geçici olabildiği gibi ürün setinin toplu üretimleri arasındaki farklar, kurulum hataları, malzeme özellikleri farkları gibi nedenler ortalamadaki bu kaymayı sabit olarak değiştirebilir.

Bu durumda tolerans analizini bu farkı dikkate alarak yapmak gerekir. Bu hata kalite ve maliyet açısından önemli bir fark yaratır. İstatiksel analize göre tasarım gereksinim sınırları sabit kaldığından onun dışında kalan hurda sayısında ciddi bir artış sağlayabilir, bu da maliyeti artırır dolayısıyla bu durumu göz ardı etmek mümkün olmayabilir. Sabit hataların etkilediği durumların yansıtılması SRSS zaman

içinde üretim süreçlerindeki hassasiyet kaybı farklı tedarikçiden alınan ürünler, değişen farklı ürünlerin toplu üretimi gibi denemeler dinamik olarak dağılımı kaydırır ve buna DRSS denir.

Tüm bunların yanında tasarım gereksinimleri göz önünde alınarak WC yönteminde olduğu gibi RSS de yeniden boyutlandırılabilir. Bazı durumlarda da süreç kapasitesini bilmediğimiz (yani satın alma yoluyla alınan parçaların) montaja dâhil edilerek kritik toleransın hesaplanması gerekebilir. Bu gibi durumda Yaklaşık (Tahmini) RSS yöntemi kullanılabilir. Bu da 0-1 arasında bir ağırlık kullanarak normal RSS ve WC modelinin bir arada kullanılması ile uygulanır [7].

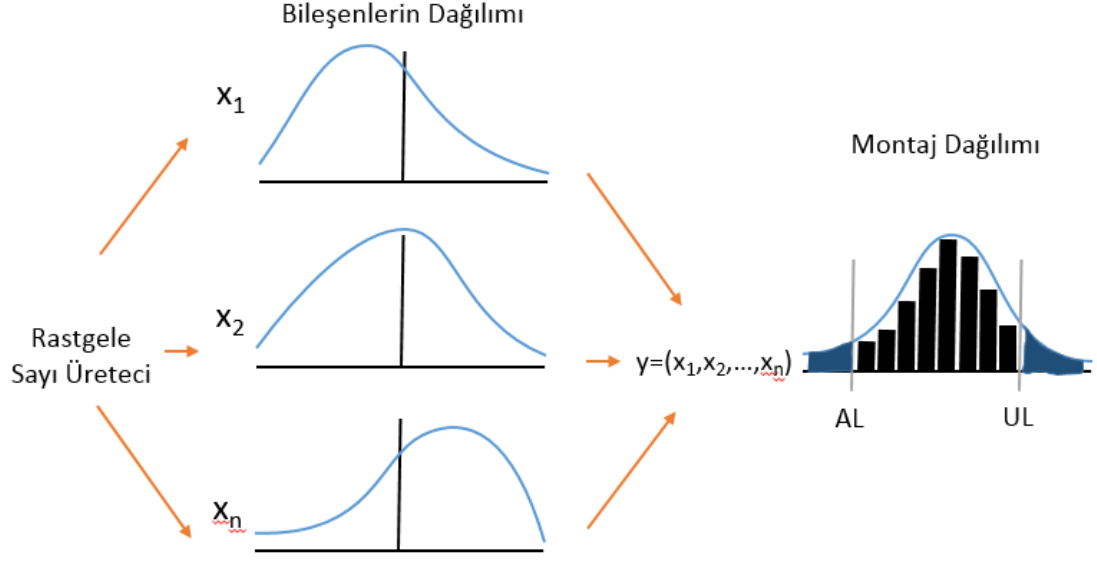
#### **2.4.2. Monte Carlo Simülasyonu**

Monte Carlo Simülasyonu (MCS) yaygın olarak fiziksel sistemlerde ve matematiksel problemlerde karşılaşılan anlaşılması basit ve sık kullanılan bir yöntemdir. Analiz kodlarıyla entegrasyonu kolay olduğundan, bilgisayar destekli çözümlerlerde ve eniyileme konularında sıkça kullanılır. Özellikle doğrusal ve doğrusal olmayan fonksiyonlarda kullanılabilmesi önemli bir avantajdır.

1940'lerden önce benzetim gerçekleştirilirken deterministik problemlere bakılarak ya da istatistikler örneklemeler yardımıyla belirsizlikler tahmin edilmeye çalışılırdı. MCS yöntemi, olasılıksal girdi ile deterministik problemlerin çözülebileceğini göstermiştir. Bu yöntem kısaca olasılıksal bir dağılım elde etmek amacıyla, tekrar eden rastgele sayısal örneklemelerin oluşturulmasına dayanır.

MCS genel olarak dört adımda gerçekleşir: 1. Girdilerin (yani sistemin değişkenlerinin) özelliklerin tespit edilmesi. 2. Belirlenen özellikler üzerinden rastgele değişkenlerin üretilmesi. 3. Bu değerlerin problem üzerinde çözdürülmesi. 4. Sonuçların toplanması ve değerlendirilmesi. Girdi özelliklerinin belirlenmesi, çıktı hata oranının azaltılmasını doğrudan etkiler. Hata oranının azaltılması benzetimdeki girdi sayısının azalmasıyla artar. Ayrıca, tekrar sayısı maliyet ile doğru orantılı

olduğu gibi problemin türüne bağlıdır ve eniyileme yapılabilir. Örneğin bu yöntemle çözülmesi analitik yöntemlerle güç olan bir matematiksel problemin her bir değişkeni için Poisson dağılımına sahip bir rastgele değer üretilerek ve problemin sonuçlarının dağılımına bakılarak sistemin karakteristiği elde edilebilir [22, 23].



Şekil 2-2 MCS ile montaj tolerans analizi

Bu çalışmada, örneklenen problem için analitik çözüm yapılabildiği durumda karşılaştırma yapılabilmesi için MCS yöntemi uygulanmıştır. Fakat çalışmanın ilgilendiği problemler ve bilgisayar destekli çözüm için bütünleştirilmesi amacıyla MCS büyük kolaylık sağlamıştır. MCS uygulanırken ve analitik çözüm yapılırken MATLAB yazılımı kullanılmıştır. MCS uygulanırken rastgele değişkenlerin üretilmesi sorunun parametrelerine bağlı olarak MATLAB kütüphanesinde bulunan fonksiyonlar ile hesaplanmıştır. Rastgele değişkenler oluşturulurken önce *rand()* (*uniformly distributed pseudorandom numbers*) fonksiyonu ile istenilen adette düzgün dağılımlı pseudo rastgele sayıları oluşturulmuş dizi, *icdf()* (*inverse cumulative distribution function*) fonksiyonu yardımıyla istenilen dağılımda ve değişkenin nominal ve tolerans değerine bağlı olarak elde edilmiştir. Örneğin, nominal değeri 9 toleransı 0.5 olan boy değeri için normal dağılımlı 100 adet rastgele imalat ölçüsü üretmek için *icdf('Normal',rand(1,100),9,0.5/3)* komutu kullanılabilir.



Problemden elde edilen sonuçların dağılımını incelerken, dağılımın ilk dört istatistiksel momentinin değerlerini elde etmek için sırasıyla MATLAB'deki *mean()*, *std()*, *skewness()*, *kurtosis()* fonksiyonları kullanılmıştır [24].

## 2.5. Vekil Model Yöntemleri

Günümüzde analitik yollarla kolaylıkla çözülemeyen, yani yüksek bilgisayar gücü gerektiren, dolayısıyla hesapsal maliyeti arttıran problemlerle karşılaşmaktadır. İstatistiksel yöntemlerin yardımıyla sonuca yakınsayan çözüm yöntemleri, geleneksel analiz kodları ya da yöntemlerin yerine sıkça kullanılmaktadır. Özellikle mühendislik alanında en sık kullanılan benzetim odaklı modelleme yöntemleri, vekil model (*surrogate model*) tabanlı yöntemlerdir bunlar metamodel olarak da bilinir. Örneğin çeşitli disiplinlerde; akış analizi, kanat tasarımı, kimyasal formülizasyon veya jeoloji uygulamalarında kullanıldığı görülmektedir. Genel olarak ifade edilen vekil model tabanlı yöntemlerin birçok yaklaşımları vardır. Bunlardan yapay sinir ağları (*artificial neural networks*), makine öğrenimi (*inductive learning*), Kriging, yanıt yüzey (*response surface*) yöntemleri sıkça kullanılanlarıdır.

Vekil model yöntemlerinin amacı, denenmiş tasarım değişkenlerinin örneklemeleri ile sistemdeki yanıtları kullanılarak bir matematiksel model oluşturmaktır. Modelin oluşturulmasında kritik öneme sahip bu deneysel noktalar, girdi değişkenlerine bağlı olup çeşitli algoritma veya deney tasarımı (*design of experiments, DOE*) yapılarak elde edilebilir. Bu deney tasarımlarından en popüler olanları Latin hiperküp örnekleme, Box-Bhenken tasarımı ve Central Composite tasarımıdır. [25]

Bu yöntem uygulanırken izlenen tipik yol şu şekilde özetlenebilir; deney tasarımıyla örneklemelerin elde edilmesi, uygun modelin seçilmesi (ki en iyi tahmin için önemlidir), bu örneklemelerin girdi olarak kullanılıp sistemin yanıtlarının alınması ve bunlar kullanılarak bir vekil modelinin oluşturulmasına dayanır. Elde edilen bu vekil model yardımıyla diğer gözlemlerin yanıtlarını tahmin etmek geleneksel yöntemlere göre avantajlı olacaktır.

Bu yöntemlerin en önemli avantajları girdi ve çıktı arasında kolay bir ilişki kurması, problemin çözümünü hızlandırması, düşük bilgisayar gücü gerektirmesi, hesapsal maliyetinin az olması, yapısı nedeniyle eniyilemeye olanak sağlaması ve gerçek analiz kodunun yerine geçerek kolaylıkla yazılımlara bütünleşebilmesidir [25, 26].

### 2.5.1. Yanıt Yüzey Yöntemi

Yanıt yüzey yöntemi (*response surface method, RSM*) istatistiksel ve matematiksel yöntemlerin birleşimiyle elde edilmiş, eniyileme formülasyonu için kullanışlı olan bir tahmin tekniğidir. Endüstride ürün veya süreçlerin, karakteristiğinin ve kalitesinin belirlenmesi için bir çözüm sunar. Bu yöntem bir matematiksel model yaratır. Bu model oluşturulurken sistemin karakteristiğini belirleyen performans ölçümleri her biri bir bağımsız değişken olan girdiler ve onun sistemdeki yanıtları arasındaki ilişki kullanılır.

RSM modelinin formu Denklem (2.3)'deki gibi varsayılır:

$$y = f(\xi_1, \xi_2, \dots, \xi_n) + \varepsilon \quad (2.3)$$

Burada yanıt fonksiyon  $y$  bilinmeyen veya çok karmaşık bir fonksiyondur.  $\varepsilon$  ise sistemin rastgele istatistiksel hatasıdır. Bu hatanın normal dağılımlı, ortalamasının sıfır ve  $\sigma^2$  varyansına sahip olduğu varsayılır. Sistemin yanıtı  $y$  değerleri kontrol edilebilir ve istatistiksel olarak bağımsız girdi değişkenleri  $(\xi_1, \xi_2, \dots, \xi_n)$ 'ne bağlıdır.

Bağımsız girdi değişkenleri  $\xi_1, \xi_2, \dots, \xi_n$  ölçülebilen ve deneysel olarak bilinen doğal değişkenlerdir. RSM uygulandığı sürece bu değişkenler  $x_1, x_2, \dots, x_n$  şeklinde isimlendirilecektir. Bu değerlerin boyutsuz, normal dağılımlı, ortalamasının sıfır ve hata ile aynı  $\sigma^2$  varyansına sahip olduğu varsayılır [27].

Yaklaşım fonksiyonunun birinci dereceden genel hali işe şu şekilde tanımlanır:

$$\hat{y} = \beta_0 + \sum_{i=1}^k \beta_i x_i + \varepsilon \quad (2.4)$$

Burada  $\beta$ 'lar parçalı regresyon sabitlerinin olduğu dizidir. Bunlar bilinmeyen değerlerdir ve regresyon analiz yöntemi olarak bilinen teknikle tahmin edilir. Aslında burada  $\beta$ 'lar her bir  $\hat{y}$  'ye karşılık gelen  $x$ 'deki varyasyonu ifade eder.

Yaklaşım fonksiyonu  $\hat{y}$  birinci dereceden kullanıldığında, doğrusal regresyon analizi olarak adlandırılır. Problemin ihtiyacına bağlı olmak üzere, daha doğru sonuç elde etmek için yüksek dereceden polinom şeklide de kullanılabilir. İkinci dereceden fonksiyonun genel hali şu şekildedir:

$$\hat{y} = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i=1}^k \beta_{ii} x_i^2 + \sum_{i < j=2}^k \beta_{ij} x_i x_j + \dots + \varepsilon \quad (2.5)$$

İkinci dereceden model kullanıldığında,  $\beta$ 'lar en küçük kareler yöntemi kullanılarak tahmin edilir. Diğer yandan  $k$  bağımsız değişken sayısını,  $y_i$  bağımsız değişkenin  $i$ 'inci gözlem değerini,  $x_{ij}$   $i$ 'inci gözlemin  $j$ 'inci bağımsız değişkenini,  $\varepsilon$  ise rastgele hatayı gösterir.

Analiz sonuçları alınırken Denklem (2.5)'e göre farklı kombinasyonlar incelenmiştir: *sabit* (1. terim) ve *doğrusal* (1. ve 2. terimler), *etkileşim* (1. 2. ve 4. terimler), *karesel* (1., 2., 3. ve 4. terimler) ve *önkaresel* (1., 2. ve 4. terimler) modelleri ele alınmıştır.

Bu durumda  $\beta$ 'lar en küçük kareler yönteminin normal denklemi çözülerek karesi alınan hataların toplam değeri en küçük değeri dikkate alınarak hesaplanır. Denklem ve değişkenleri birer matris olarak aşağıda verilmiştir [28,29]:

$$\beta = [X'X]^{-1} X'\hat{Y} \quad (2.6)$$

Burada  $X$  tasarım matrisi (yani örneklem veri noktaları),  $X'$  matrisin transpozu,  $\hat{Y}$  ise her bir örnekleme karşılık gelen yanıtları içeren sütun vektörüdür.

### 2.5.2. Kriging Yöntemi

1950'lerin başlarında Güney Afrika'da maden mühendisi olan Daniel G Krige örneklemelerin uzaysal bağımlılığına dayanan ilk interpolasyon yöntemini yayınlamıştır. Daha sonra 1970'lerde Fransız matematikçi George Matheron bu çalışmayı daha pratik hale getirerek kullanılabilir kılmış ve Kriging metodu (KM) olarak adlandırmıştır. Böylece geoistatistiğin temelleri atılmıştır. Günümüzde mühendislik alanlarında yeni yeni kullanılan bir yöntem olarak karşımıza çıkar. Az kullanılmasının en büyük nedeni kısıtlı hazır paket programların azlığıdır.

Bu teorinin temelinde aslında gözlemlenen değerler ve yerelde buna yakın değerlerin uzaklığı kullanılarak ağırlıklandırılır ve tahminler için kullanılır. Bu teori stokastik ve deterministik bileşenleri birarada kullanır. Deterministik bileşenler genel veya yerel eğilimler olabilir. Stokastik bileşenler ise rastgele ve otomatik olarak eşleştirilmiş parçalardan oluşur [30,31].

Modeli oluşturmak için ağırlık sabitlerinin belirlenmesi büyük önem taşır, böylece variogram oluşturulur ve model uydurulabilir. KM genel şekli ile aşağıda verilmiştir:

$$\hat{y}(x) = \sum_{i=1}^k \beta_i f_i^T(x) + Z(x) \quad (2.7)$$

Burada ilk terim eğilim modeli diğer terim  $Z(x)$  ise gaussian dağılımlı, ortalaması sıfır ve  $\sigma^2$  varyansına sahip olduğu varsayılan stokastik süreçtir ya da sistematik sapma da denebilir. Burada  $\hat{y}(x)$  evrensel olarak tasarım uzayına yaklaştığında,  $Z(x)$  yerelleşmiş sapmaları oluşturur böylece KM  $n$  sayıda örneklem veri noktası için interpolasyon yapar.

$Z(x)$ 'in yakınsama matrisi şu şekildedir:

$$\text{Cov}[Z(x_i), Z(x_j)] = \sigma^2 R(x_i, x_j) \quad i, j = 1 \dots n \quad (2.8)$$

Burada  $\sigma^2$  süreç varyansı,  $R(\dots)$  korelasyon modellerdir ve üstel, normal, küresel, ve doğrusal olabilir. Bu çalışmada aşağıda verilen gaussian modeli standart olarak kullanılmıştır.

$$R(x_i, x_j) = \exp\left(-\theta_k \sum_{k=1}^n |x_i^k - x_j^k|^p\right), 0 < p \leq 2 \quad (2.9)$$

Burada modeli uydurmak için kullanılan bilinmeyen korelasyon parametreleri  $\theta_k$  olarak belirtilmiştir.  $x_i$  ve  $x_j$  ler örneklemelerin  $k$ 'nci bileşenleri ve  $k$  ise tasarım değişkenidir.

KM yönteminin en zaman alıcı parçası ise modellemedir. Modelleme sırasında eğer örneklem noktaları birbirine çok yakınsa korelasyon matrisi tekil çıkabilir. Bu nedenlerle örneklemelerin seçimi önemlidir. Eğer belli bir deney tasarımı için deneysel veri noktaları şu şekilde ifade edilirse;

$$x = (x_1, x_2, x_3, \dots, x_n) \quad x_i^k \in \mathfrak{R}^p \quad (2.10)$$

Burada  $p$  tasarım değişkenlerinin sayısı dolayısıyla  $x$  ise  $n \times p$  tasarım matrisi olur. Sonuç olarak yanıt fonksiyonu aşağıdaki matris şeklinde ifade edilebilir;

$$Y = \{y_1(x), y_2(x), \dots, y_n(x)\} \quad (2.11)$$

Modelin parametrelerini oluşturan  $\beta$  vektörü ve varyans tahmini  $\sigma^2$  aşağıdaki şekilde hesaplanır:

$$\hat{\beta} = (F^T R^{-1} F)^{-1} (F^T R^{-1} y) \quad (2.12)$$

$$\sigma^2 = \frac{1}{n} (y - F\hat{\beta})^T R^{-1} (y - F\hat{\beta}) \quad (2.13)$$

Burada  $F$  her bir hesaplanan  $f(x)$  fonksiyonlarını içeren matris ve  $R$  korelasyon matrisidir.  $\beta$  ve  $\sigma^2$ , denklemdeki  $\theta_k$  için bilinmeyen korelasyon fonksiyon parametrelerinin en fazla olasılıklı tahminidir (*maximum likelihood estimation*). Bu da  $1/2 [N \ln \hat{\sigma}^2 + \ln \det(R)]$  fonksiyonunun minimumu bulunarak hesaplanır [31].

Regresyon modeli doğrusal, ikinci ve üçüncü dereceden olabilir. Bu çalışmada tüm bu hesaplamalar DACE araç kutusu MATLAB üzerinde çalıştırılarak yapılmıştır [32,33].

## 2.6. Bilgisayar Destekli Yazılımlarla Tolerans Analizi

Günümüz endüstrisinde, üretimden teste ve tasarıma kadar birçok aşama, bilgisayar destekli hesaplamaların yapıldığı ve süreçlerin analiz edildiği bir ortamda sürdürülür. Ürünler henüz “ilk örnek” aşamasında, üretime geçmeden önce modellenerek birçok yönden incelenir ve artık mühendisler toleransları da bilgisayar desteği ile analiz etmeye başlamıştır. Artık her firmanın ürün modellerinin bilgisayar destekli tasarım (*Computer Aided Design, CAD*) olarak sayısal modelinin bulunması kaçınılmazdır. Dolayısıyla tolerans analizinin, CAD model bilgilerini kullanarak uygulanması çok daha pratik bir yoldur.

Genel olarak mekanik tolerans analizi yazılımları, bilgisayar destekli toleranslama (*computer aided tolerancing, CAT*) olarak adlandırılır. Bu yazılımlar sadece süreç ve analizi değil sentez gibi tolerans tasarım aktivitelerini içerir. Doğal olarak tolerans sunumu, gereksinimleri ve standartlarında teorilerini kapsar. Fakat her biri arka planda aynı teoriyi kullanmamaktadır. Diğer yandan geliştirilen çeşitli tolerans analiz

teorilerinin çok karmaşık ve el hesaplarını genellikle çok zor olmasının yanında, bazılarını (örneğin T-Map) ise elle çözmek imkânsızdır. Mümkün olan durumda ise montaj elemanlarının içerdiği yapı ve sayıları nedeniyle çözüm, içinde çıkılmaz bir hal alabilir. Bu nedenlerle CAD destekli CAT yazılımları kullanmak olmazsa olmazdır.

Ticari olarak satılan farklı tolerans analiz yöntemlerine dayanan CAT yazılım paketleri tek başına (*stand-alone*), eklenti olarak (*add-on*) ya da uygulama tablosu (*spreadsheet*) olarak çalışan şekliyle de sunulmaktadır. Bu eklenti paketleri günümüzde kullanılan ticari CAD yazılım firmaları tarafından üretilen popüler CATIA, Unigraphics (UG), Solidworks, Pro/Engineer gibi CAD yazılımlarıyla uyumludur.

Ticari olarak çok satan ve sıkça kullanılan CAT yazılımları; CETOL 6 Sigma™ (üreticisi Sigmetrix LLC, dünya dağıtıcısı RandWorldwide ve Parametric Technology), eM-TolMate™ (üreticisi Tecnomatix Technologies Ltd), Unigraphics yaratıcılarından VSA-GDT™ ve VSA.3D™ (üreticisi Engineering Animation Inc. EAI), CATIA yaratıcılarından 3DCS™ (üreticisi Dimensional Control Systems Incorporated, DCS) gibi sıralanabilir. Yukarda bahsedilen yazılımlar hem WC hem de istatistiksel yöntemler kullanılarak analiz yapabilirler. 3DCS Monte Carlo benzetimi de analiz yapabilir. VSA üç boyutlu nokta model kullanarak model parametrelerini değiştirerek varyasyonu inceler. Her bir parametreye uygulanan değişim ile diğerleri arasındaki ilişkiyi çıkarır. Daha sonra doğrusal tolerans yığılması kullanılarak ya da MCS ile tolerans analizi yapar. En çok kullanılan 3DCS ve VSA montaj dağılımı, her bir parçanın katılımını ve hurda sayısını da hesaplayabilir [9, 34].

Bu yazılımlar birçok açıdan kolaylık sağlasa da, bazı dezavantajları mevcuttur. Tüm bu yazılımların, en çok kullanılan tolerans sunum standardı olan ASME Y14.5 ile %100 uyumlu oldukları söylenemez. Her biri 2 ve 3 boyutlu analizlerde tam olarak çalışmasa da farklı yollar uygulanarak en iyi tahmini bize sunarlar. Ayrıca bu

yazılımların kullanılması katı yordamlara bağılı olduğu için kullanıcı açısından eğitim gerektirmektedir. Tüm bunların sonunda genelde istatistiksel olarak elde edilen sonuçların yorumlanması ise tasarımcının yetkinliğine kalmaktadır. Bunlar da parçanın fonksiyonunu bilmek, üretim süreçlerine hâkim olup gerçekçi olansları tahmin etmek ve kalite kontrol yönteminin nasıl olduğunu bilmek ile ilgilidir.

### **2.6.1. TolAnalyst**

Bahsedilen yüksek maliyetli yazılımların dışında, 2008'den bu yana kullanılan TolAnalyst eklentisi ise orta ölçekli firmalarca yaygın olarak kullanılan sektördeki lider Dassult Systems tarafından geliştirilmiş Solidworks üzerinde çalışır. TolAnalyst™ toleransların ve montaj yöntemlerinin bir montajın iki unsuru arasındaki ölçümlendirmeye ilişkili yığılmaya olan etkilerini incelemek için kullanılan bir tolerans analiz aracıdır [34]. Bu çalışmada maliyeti düşük ve yaygın kullanımı olması nedeniyle Solidwork yazılımı kullanılmıştır. Solidworks, uygulama programlama arayüzü (*Application Programming Interface, API*) desteklemesi nedeniyle model üzerinde çalışan otomatikleştirilmiş uygulamalar geliştirilmiş ve bu çalışma kapsamında üretilen, tolerans tekniği uygulanabilir kılınmıştır.

TolAnalyst uygulaması tolerans analiz yöntemlerinden tolerans yığılması ile WC ve RSS uygulayabilirken MCS yöntemini uygulayamaz, deformasyon, sıcaklık ve yerçekimi etkilerini dikkate alamaz. Bu yöntemleri kullanılarak bunların en küçük ve en büyük değerlerini hesaplayabilir ve görsel olarak gösterebilir. En yüksek malzeme durumu (*Maximum Material Condition, MMC*) toleransları ile yerleşim toleranslarından açısallık, paralellik diklik ve bazı açı boyutlandırmaları dikkate alırken unsurun oryantasyonunu örtülü olarak etkileyen konum, yüzey profili, doğrusal ölçümlendirmeleri dikkate almaz. TolAnalyst'in çalışabilmesi için CAD modelin yine Solidworks içinde yer alan geometrik ya da boyutsal toleranslamaya uygulaması olan DimXpert™ ile otomatik ya da el ile tam olarak girilmesi gerekir. Dolayısıyla TolAnalyst, DimXpert'in desteklemediği; küreselin boyutlandırılması, kavisli yüzeyler, kesişim yüzeyleri, fillet ve chamfers ile bunlardan oluşan unsur ve



ölçülendirmeleri desteklemez. DimXpert dolayısıyla TolAnalyst, GD&T şema standardı olarak ASME Y14.5-2009 ve ISO 16792:2006 ile uyumludur.

TolAnalyst kullanılırken temel olarak şu yöntem izlenir. Ölçümün alınması: model DimXpert yardımıyla tamamiyle ölçülendirilir. Ölçülendirme geometrik ve boyutsal olabilir. Model TolAnalyst ile açılarak analiz edilecek kritik ölçü belirtilir. Bu kritik boşluğun doğrusal bir uzaklığıdır. Montaj sırasının belirlenmesi: imalat sırasına göre montaj sıralaması belirlenir. Bu işlem tolerans zincirini oluşturmak için önemlidir ve montajın basit sırasını oluşturur. Montaj sınırlandırması: bileşenlerin geometrik şekilleri esas alınarak montaj sıralamasına nasıl yerleşeceğinin sınırlandırmalarını içerir. Bu aşama sonuçları etkileyen önemli bir adımdır. Daha sonra sonuçlar alınır ve analiz edilir. Bu aşamada nominal değer, WC ve RSS için en küçük ve en büyük değerler alınabilir ve grafiksel olarak izlenebilir. Bunun yanında hesaplama katkı sağlayan ölçülerin neler olduğu ve katılımları da yüzde olarak incelenebilir. Örneğin basitçe; nominal değer 10 mm, maksimum en kötü senaryo koşulu sonucu 11 mm ise ve unsur 1 mm'lik tolerans yığılmasının 0,1'ini tüketiyorsa; unsur katılımı %10'dur.

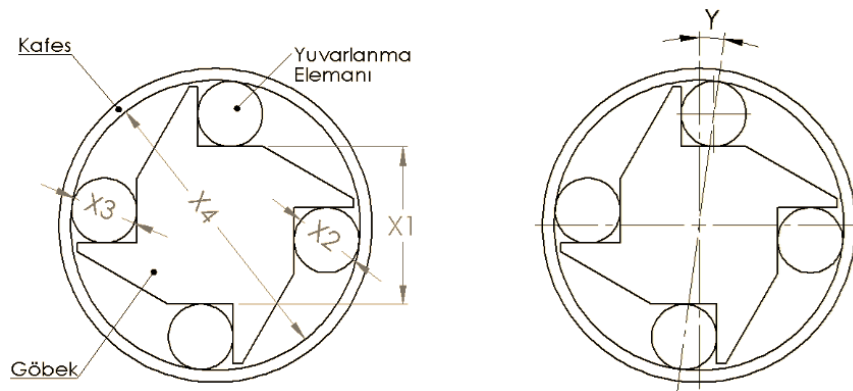
Bunun dışında, oryantasyon toleranslarını kullanmak, orijin unsuruna dik kullanmak, pim ve bağlantı elemanlarını yüzdürme seçime bağlıdır. "*Oryantasyon Toleransları*" seçeneği TolAnalyst'in unsur oryantasyonunu açık bir şekilde kontrol eden toleransları dikkate almasını sağlar. "*Pim ve bağlantı elemanlarını yüzdür*" seçeneğiyle birleştirildiğinde, "*Oryantasyon Toleransları*" seçeneği TolAnalyst'in parçaları en kötü durum koşullarına döndürmek için unsurlar arasındaki boşlukları kullanmasını sağlar. Orijin unsuruna dik seçeneği, ölçümün yönünü ölçümün "*Buradan Ölç*" unsuruna dik olacak şekilde yeniden yönlendirir. TolAnalyst en kötü durum tolerans koşullarını hesaplarken sabit ve yüzen bağlantı elemanlarından doğan boşluk paylarını göz önünde bulundurabilir [35].

### 3. BİR KAVRAMA ELEMANININ TOLERANS ANALİZİ (2 BOYUTLU PROBLEM)

#### 3.2. Problemin Tanımı

Bu çalışmada kullanılan kavrama montajı ilk olarak Fortini tarafından kullanılmış [10] ve onun ismiyle anılmaktadır (*Fortini's Clutch Assembly*). Daha sonra tolerans analizi olarak Greenwood ve Chase [11] ile ekibi bu problemin tolerans analizi üzerine çalışmalar yapmıştır. Bu tolerans yığılması uygulaması için seçilen montaj elemanının, iki boyutlu bir tasarım fonksiyonuna sahip olması, fonksiyonun doğrusal olmayan türden olması ve parametrelerin boyutsal toleranslar içermesi nedeniyle iyi bir örnektir.

Kavrama elemanı Şekil 3-1'de ifade edildiği gibi üç bileşenden oluşur. Bunlar kafes, yuvarlanma elemanı ve göbektir. Bu montaj tek yönde hareket etmek üzere tasarlanmıştır. Göbek bir shaft yardımıyla bir mekanizmaya bağlı olabilir. Saat yönünde döndüğü sürece yuvarlanma elemanları kafes içinde hareket eder. Aksi yönde ise yuvarlanma elemanları kafes ve göbek arasında sıkışır ve kilitlenir.



Şekil 3-1 Fortini'nin Kavrama Montajı

Bu analizin asıl amacı yuvarlanma elemanının konumunun kafes ve göbeğe göre tam olarak belirlenmesidir. Montajın en iyi performansı gösterebilmesi için yuvarlanma

elemanının konumu çok sınırlı aralık içerisinde. Bu sınır  $\alpha$  olarak adlandırılan bir açı olan bağımlı değişkene bağlıdır. Bu  $\alpha$  açısı kritik ölçü olarak da tanımlanır. Kritik açı  $\alpha$  nin tasarım sınırları  $0.122 \pm 0.035$  radyan (  $7.0 \pm 2.0$  derece ) olması beklenir. Bu açı aşağıdaki Denklem (3.1)'de verildiği gibidir ve y tasarım fonksiyonu olarak adlandırılır.

$$y = \arccos \left[ \frac{x_1 + \frac{(x_2 + x_3)}{2}}{x_4 - \frac{(x_2 + x_3)}{2}} \right] \quad (3.1)$$

Kritik ölçü  $y$  Şekil 3-1'de ne olduğu gösterilen ve  $x_1, x_2, x_3, x_4$  olarak adlandırılmış her biri bağımsız olan rassal değişkenlere bağlıdır. Bahsedilen rassal değişkenlerini her birinin normal dağılıma sahip olduğu varsayılmış ve ortalama değerleri ile standart sapmaları Çizelge 3-1'de verilmiştir.

Çizelge 3-1 Rassal değişkenlerin ortalama ve standart sapmaları

Bağımsız değişkenler	Ortalama	Standart Sapma
$x_1$	55.29	0.10160
$x_2$	22.86	0.01016
$x_3$	22.86	0.01016
$x_4$	101.69	0.20320

### 3.3. Geleneksel Çözüm

Geleneksel çözüm olarak ifade edilen tolerans yığılma analizi yöntemleri olarak WC ve RSS kullanılmıştır. Bunun yanından benzetim yöntemi olarak MCS de uygulanmıştır. Tüm bu yöntemler MATLAB yazılımı kullanarak uygulanmıştır.

Hesaplamalar yapılırken MATLAB yazılımında sembolik fonksiyon özellikleri kullanılmıştır. Bunun için önce değişkenler sembolik olarak tanımlanmıştır. Ardından tasarım fonksiyonu  $y$  tanımlanmıştır. Fonksiyon *subs* komutu yardımıyla

her bir deęişkenin nominal deęeri yerine konularak çözdürülmüştür. Böylece radyan cinsinden  $y$  açısı elde edilmiş ve bu *radto deg* komutuyla dereceye çevrilmiştir. Böylece nominal  $\alpha$  açısı elde edilmiştir.

WC ve RSS yönteminde ihtiyacımız olan her bir deęişkenin hassasiyetinin hesaplanması gerekir. Her bir deęişkenin hassasiyeti (yani kritik açı üzerindeki etkisi) fonksiyonun türevi alınarak hesaplanabilir. MATLAB kullanarak hesaplandığından program içine gömülü olan *diff* komutu kullanılmıştır. Tüm toleranslar  $3\sigma$  kabul edilmiştir. WC ve RSS kullanılırken Denklem (2.1) ve (2.2) kullanılmıştır. Tüm bu hesaplamalar EK 1’de verilen “*rss\_mcs\_clutch.m*” dosyasındaki kodlar ile hesaplanmıştır.

MCS yöntemi uygulanırken de daha önceki bölümde bahsedildiği gibi MATLAB yazılımı içerisindeki gömülü fonksiyonlardan faydalanılmıştır. Kod, fonksiyona benzetim sayısı  $n$  girilerek çalışmaktadır. Fonksiyon çalıştıktan sonra her bir durum için üretilen deęişkenlerin bulunduğu  $n \times 4$  büyüklüğünde bir matris ve bunlara karşılık gelen kritik açıların bulunduğu  $1 \times n$  büyüklüğünde bir vektör verir. Fonksiyon içerisinde ilk önce deęişkenler tanımlanmıştır; bunlar deęişkenlerin nominal deęerleri ve tolerans deęerleridir. Toleranslar  $3\sigma$  kabul edilmiştir. Ardından *icdf()* (*Inverse cumulative distribution function*) fonksiyonu yardımıyla örnek problemde kabul edildiği üzere normal dağılımlı verilen nominal ve toleranslara sahip bağımsız deęişkenlerin rastgele deęerleri üretilmektedir. Bunun yanında tasarım fonksiyonu girilerek kritik açıları hesaplanmaktadır. Tüm bunlar hesaplandıktan sonra sonuçların ilk dört istatistiksel momenti de hesaplanır. Ardından tasarım sınırlandırması olarak verilen tolerans deęeri tanımlanarak dışımda kalan hurda adet ve yüzdeleri hesaplanır. Son olarak da deęerlerin histogramı hesaplanarak çıktısı alınır. Tüm bunlar EK 2’de verilen “*mcs\_clutch.m*” dosyası ile hesaplanmıştır.

### 3.4. Vekil Model Aracılığıyla Çözüm

Bu problem üzerinde yanıt yüzey yöntemi ve Kriging olmak üzere iki tip vekil model yöntemi uygulanmıştır. Bu iki yöntem uygulanırken tek bir veri seti kullanılmış, böylece karşılaştırma yapılırken rastgele üretilen değişkenlerden oluşabilecek farklılıkların önüne geçilmesi sağlanmıştır. Yöntemlerde fonksiyonu oluşturmak için kullanılan veriler de bu set içinden ilk üretilenler seçilmiştir.

RSM kullanılırken izlenen yol şu şekilde özetlenebilir. Öncelikle MCS yöntemiyle yüz bin adet değişken oluşturulmuştur. Bunlar  $s$  dizisi üzerinde her bir  $x$  değişkeni olmak üzere dört sütun olarak saklanmıştır. Bu değişkenler problemin fonksiyonu kullanılarak sonuçları alınmış ve  $y$  dizisi üzerine saklanmıştır. Bu işlem için EK 3'deki "mcs\_vekil\_clutch.m" fonksiyonu kullanılmıştır.

Daha sonra  $Nmodel$  olarak belirtilen sayıda veri setinden değişkenler çekilmiş ve *regstats* komutu yardımıyla model oluşturmak için kullanılmıştır. Daha sonra bu fonksiyonun  $\beta$  katsayıları  $b$  değişkeni üzerine alınmıştır. Ardından model oluşturma bölümü bitirilmiş ve ardından yanıt yüzey yöntemi ile tahmin oluşturma bölümüne geçilmiştir. Burada elimizdeki fonksiyon yardımıyla az sayıda benzetimle, ya da deneysel çalışma (ki bu "ilk ürün" ürünlerin sonuçları da olabilir) ile sistemin karakteristiği elde edilmeye çalışılmıştır. Sonuçta benzetim için problemin 4 değişkeni için tekrar rastgele değerler oluşturulmuş ve sistem yanıtının tahmini için kullanılmıştır. Tüm bu hesaplamalar EK 4' deki "rsm\_clutch.m" dosyası ile yapılırken fonksiyon excel dosyasına otomatik olarak sonuçları kaydeder. "rsm\_clutch.m" fonksiyonunu içerisinde kullanılan ve dağılımın momentleri ile tasarım sınırları dışında kalanların hesaplandığı fonksiyon "statistic.m" EK 5' de verilmiştir.

Yukarıda bahsedilen benzetim çalışmaları 20, 40 ve 100 adet örneklem noktası kullanılarak tekrarlanmıştır. Kritik açı  $y$  tahmini için ise 1000, 10,000 ve 100,000 benzetim yapılmış ve fonksiyona uygulanmıştır. Bu fonksiyonlar her bir deneme veri

seti için doğrusal ve ikinci dereceden olmak üzere fonksiyonun kesişim (*interaciton*) kısmı ile önkaresele (*prequadratic*) kısmı alınarak hesaplanmıştır. Tümü için normal dağılımlı bağımsız değişkenler kullanılmıştır. RSM için hesaplanan tüm sonuçların tablosu EK 7’de verilmiştir.

Kriging yöntemi için Hans Bruun Nielsen [1] öncülüğünde hazırlanan DACE (*Design and Analysis of Computer Experiments*) isimli araç kutusu kullanılmıştır. Bu yöntemde de RSM için kullanılan veri seti kullanılmıştır. Belirlenen tasarım verisi, regresyon ve kolerasyon modeli için oluşturması için *dacefit* fonksiyonu kullanılmıştır. Yazılan kodda *s* olarak belirtilen model için kullanılan tasarım verisi (bu çalışmada sabit veri setinin ilk 20, 40 ve 100 adedi olarak kullanılmıştır), *y* olarak belirtilen ise bu verilere karşılık gelen problemin yanıtıdır. Regresyon modeli olarak 0, 1 ve 2. dereceden polinomların üçü de denenmiştir. Korelasyon modeli olarak ise "*corrgauss*" komutuyla Gaussian dağılımı kullanılmıştır. Ardından MCS de her bir parametre için 1000, 10,000 ve 100,000 adet değişkenle tahminler yapılmıştır. Kullanılan kod tüm hesaplamaları excel dosyası üzerine kaydeder ve RSM de olduğu gibi "krig\_clutch.m" dosyası yardımıyla değişkenleri oluşturur. Tüm kodlamalar EK 6’de verildiği gibidir. Kriging için hesaplanan tüm sonuçların tablosu EK 8’de verilmiştir.

### **3.5. Bulgular**

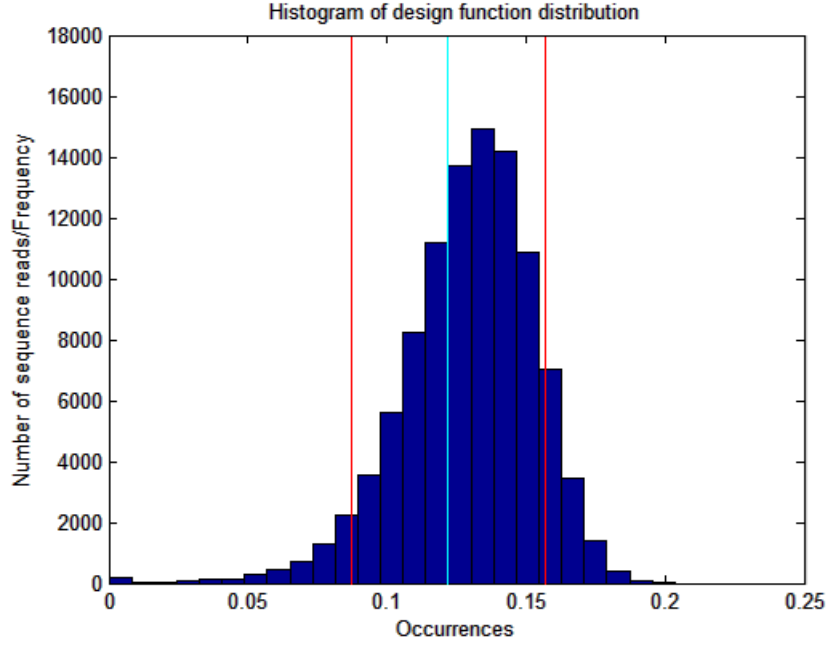
Önceki bölümde bahsedilen yöntem ve çözümler aracılığı ile elde edilen sonuçlar şöyledir. Problemden geleneksel yöntemlerden olan WC ile çözüldüğünde, tolerans değerinin 0,0339 olduğu dolayısıyla nominal değerle uyumlu olduğu gözlenmiştir. Kritik boşluğun en küçük değerinin 0,0881 olduğu, en büyük değerinin 0,1559 olduğu bulunmuştur. Neyse ki tasarım kriteri olan 0,035 olan tolerans içerisinde kaldığından en kötü durumda bile verilen toleransların uygun olduğu toleransların herhangi bir yeniden boyutlandırmaya gerek olmadığı görülmüştür. RSS’ye göre standart sapma 0,0237 ve elde edilen sonuç da WC’e göre daha muhafazar olup en

küçük deęerin 0,0978 en büyük deęerin ise 0,1452 olduęu görülmüştür. Ortalama deęer ise sembolik fonksiyonla çözülmüş ve 0,1215 radyan deęerindedir.

MCS ile 1000, 10,000, 100,000 örneklem için benzetim yapılmış ve sonuçlar tabloda verilmiştir.Çizelge 3-2’de görüldüğü üzere 10,000 örneklem kabul edilebilir olmasına rağmen bu soru için 100,000 de sonuca yakınsamıştır. 100,000 deęerle yapılan benzetim için histogramı Şekil 3-2’de gösterilmiştir. Şekildeki kırmızı çizgiler tasarım sınırları, mavi ile gösterilen ise y deęerinin nominal deęerini gösterir.

Çizelge 3-2 Fortini'nin kavrama montajı için MCS sonuçları

Kontrollü Montaj Deęişkenlerinin Çıktıları / Benzetim Sayısı	1000	10000	100000	1000000
Ortalama [radian]	0,1303	0,1287	0,1293	0,1293
Standart Sapma	0,0231	0,0245	0,0238	0,0236
Skewness	-0,5961	-0,8513	-0,7185	-0,7316
Kurtosis	3,5806	4,6820	4,0620	4,2006
Üst Sınır Hurdası [%]	11,3000	10,0900	10,0430	9,9619
Alt Sınır Hurdası [%]	4,3000	5,4100	4,7020	4,6510
Toplam Hurda[%]	15,6000	15,5000	14,7450	14,6129
Hesaplama zamanı [saniye]	0,0441	4,0618	8,1358	12,6253



Şekil 3-2 Fortini'nin Kavrama Montajı için 100,000 örneklemlili MCS histogramı

Çizelge 3-4'de verilen ve RSM ile elde edilen tolerans analizi sonuçları ile Çizelge 3-2'de verilen MCS sonuçları ile karşılaştırıldığında, RSM'de karesel model kullanıldığında en yakın sonuçların elde edildiği görülmektedir. Ayrıca Çizelge 3-2 ile EK 7'de verilen RSM sonuçları karşılaştırıldığında, RSM oluşturulurken 40 adet örneklem noktası kullanılmasının uygun olduğu görülmektedir.

Çizelge 3-3 Fortini'nin kavrama montajına uygulanan 40 örneklemlili RSM sonuçları

Regresyon Tipi	doğrusal	etkileşim	karesel	önkaresel
Ortalama [radyan]	0,1304	0,1307	0,1296	0,1302
Standart Sapma [radyan]	0,0245	0,0247	0,0226	0,0235
Skewness	-0,0375	-0,0234	-0,6838	-0,3555
Kurtosis	2,9850	3,0006	3,6737	3,1807
Üst Sınır Hurdası [%]	13,78	14,2	9,5	12,24
Alt Sınır Hurdası [%]	3,83	3,79	4,38	4,1
Toplam Hurda[%]	17,61	17,99	13,88	16,34
Hesaplama zamanı [saniye]	0,11331	0,1021	0,09154	0,09927
Tahmin	10000	10000	10000	10000



Benzer şekilde Çizelge 3-4’de gösterilen ve Kriging yöntemiyle elde edilen sonuçlar ile Çizelge 3-2’de verilen MCS sonuçları ile karşılaştırıldığında, KM’de karesel model kullanıldığında (*regpoly2*) en yakın sonuçların elde edildiği görülmektedir. Ayrıca Çizelge 3-2 ile EK 8’de verilen KM sonuçları karşılaştırıldığında, KM oluşturulurken de RSM’e benzer şekilde 40 adet örneklem noktası kullanılması gerektiği görülmektedir.

Çizelge 3-4 Fortini’nin kavrama montajına uygulanan 40 örneklemlili KM sonuçları

Regresyon Tipi	regpoly0	regpoly1	regpoly2
Ortalama [radyan]	0,1189	0,1305	0,1296
Standart Sapma [radyan]	0,0067	0,0244	0,0226
Skewness	0,2824	-0,0764	-0,6835
Kurtosis	7,3471	3,0709	3,6900
Üst Sınır Hurdası [%]	0,01	13,45	9,56
Alt Sınır Hurdası [%]	0,08	4,04	4,37
Toplam Hurda[%]	0,09	17,49	13,93
Hesaplama zamanı [saniye]	0,18473	0,18754	0,19500
Tahmin	10000	10000	10000

Yukarıdaki sonuçlar incelenirken benzetim sayılarının aynı olması ve vekil model için kullanılan tasarım değişken sayılarının aynı olması dikkate alınmıştır. MCS ile elde edilen sonuçlar referans alındığında; 4 değişkenli kavrama montajı problemi için vekil model oluşturulurken kullanılan örneklem noktası adeti 20 olduğunda yetersiz, 40 olduğunda ise yakınsak sonuç vermiştir. Her iki vekil model yönteminde karesel polinom kullanımı en yakın sonucun alınmasını sağlamıştır. Hesaplama zamanı olarak ise vekil model yöntemleri en hızlı sonucu verdiği düşünülebilir. Genel olarak problemin yapısı gereği birbirini doğrulayan çok yakın sonuçlar elde edilmiş ve uygulamanın geçerli olduğu ve çok daha karmaşık problemlerde uygulanarak incelenmesi gerektiği sonucuna ulaşılmıştır.

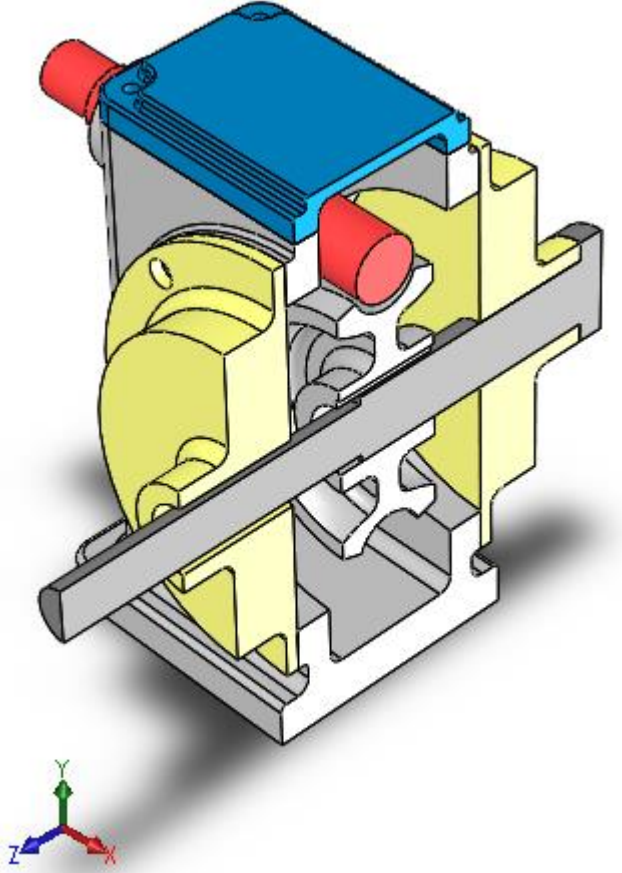
## 4. BİR DİŞLİ KUTUSUNUN TOLERANS ANALİZİ (3 BOYUTLU PROBLEM)

### 4.2. Problemin Tanımı

Bu çalışmada önerilen analiz yöntemi önceki probleme uygulanmasının yanında hataların ve olasılıkların daha iyi irdelenmesi, farklı parametrelerin karşılaştırılarak ve bilgisayar destekli yazılımlar ile yöntemin bütünleştirilmesi amaçlarıyla yeni bir örnekte denenecektir. Kullanılacak örnek üç boyutlu olması ve içerisinde boyutsal toleranslara ek olarak geometrik toleransları da barındırması ve tolerans analiz programı için hazır olması nedeniyle seçilmiştir.

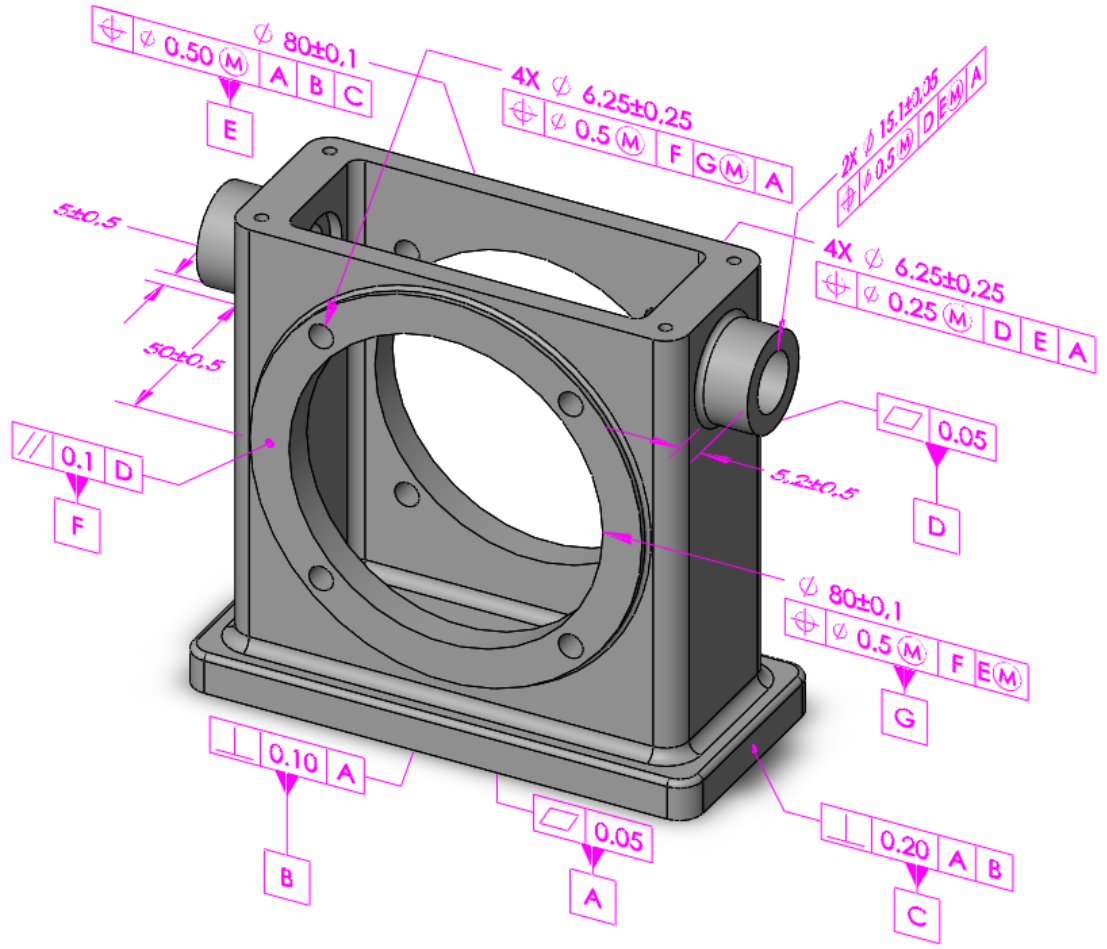
Seçilen problem Solidworks yazılımının 2014 sürümü içerisinde tolerans analizi öğretici seti içerisinde standart olarak sunulan bir katı montaj modelidir. Bu model var olan haliyle GD&T içermesi ve program içerisinde yer alan tolerans analizi eklentisi TolAnalyst ile uyumlu olması nedeniyle avantajlıdır. Bunun yanısıra Solidworks yazılımı, bazı işlemleri otomatikleştirmek için uygulama geliştirme desteği (API) sağladığı için bu çalışmada kullanılacak yöntemleri hızlı bir şekilde program ile bütünleştirmeye ve otomatik çözmeye imkan vermesi açısından çok faydalıdır.

Bu model bir dişli kutusudur ve bir yönde elde edilen hareketi diğer yöne aktarır. Bu montaj elemanı bir muhafaza (*Housing*), iki yuvarlak kapak plakası (*Round Cover Plate*), bir sonsuz dişli şaftı (*Worm Gear Shaft*) ve bir dişli çark (*Worm Gear*), bir ters şaft (*Offset Shaft*) ve bir dikdörtgen kapak plakası (*Cover Plate*)'dan oluşur. Bu elemanlar ve montajı Şekil 3-3'de gösterilmiştir.

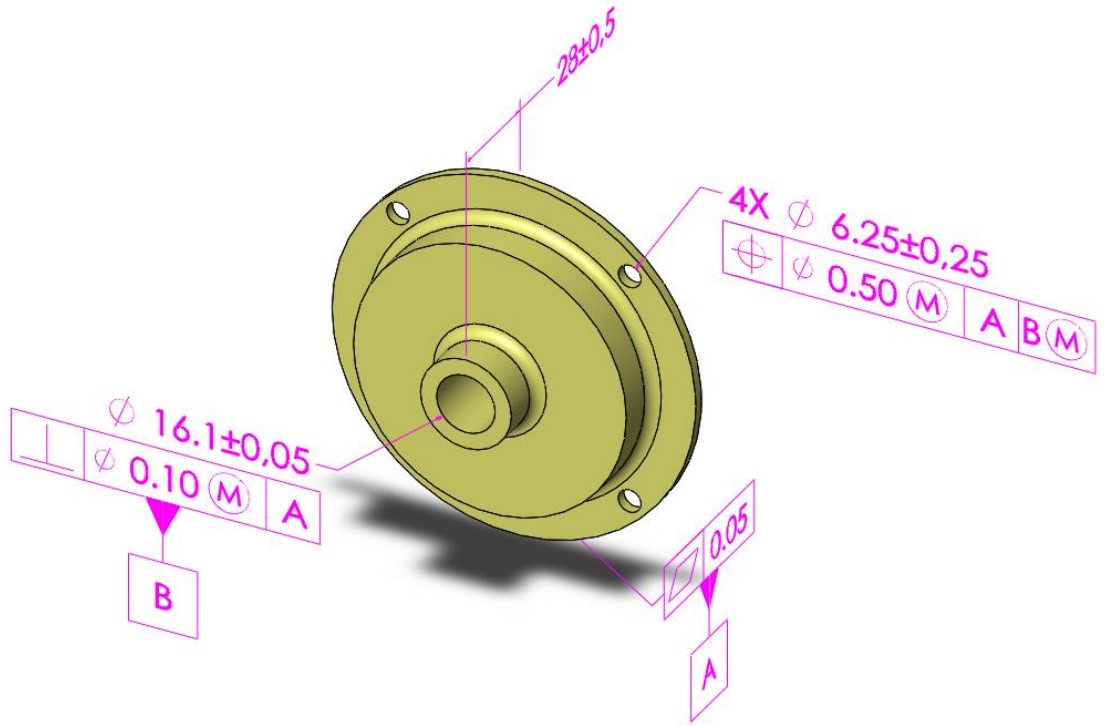


Şekil 3-3 Dişli Kutusu Montaj elemanlarının görüldüğü kesit

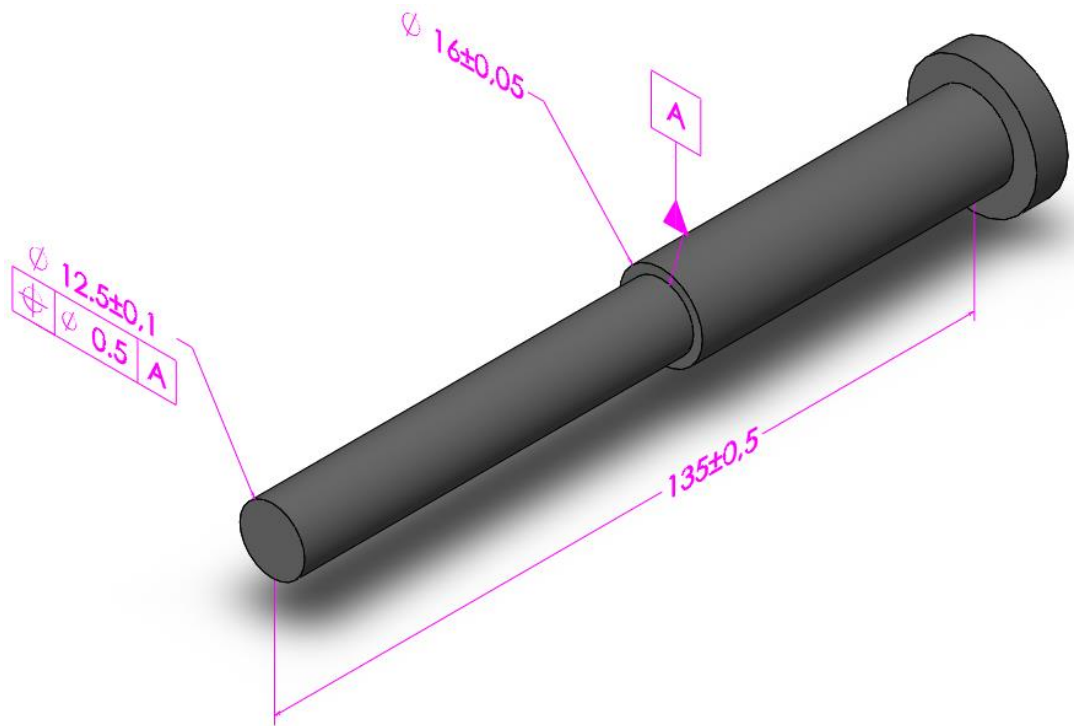
Bu analizde incelenilen kritik ölçü ise sonsuz dişli şaftı ile ön tarafta bulunan yuvarlak kapak plakası deliği arasında kalan en düşük boşluktur. Bu boşluğun tasarım sınırları problem içinde verilmemiş olmakla beraber, bileşenlerle çakışma olmaması ve fiziksel olarak montajlanabilme durumu bir gereksinim olarak belirlenmiştir. Analizde ihtiyaç duyulan parçanın GD&T toleranslamaları, Şekil 3-4 muhafaza elemanı için, Şekil 3-5'te yuvarlak kapak plakası için ve Şekil 3-6'da sonsuz dişli şaft için gösterilmiştir.



Şekil 3-4 Muhafaza elemanınının geometrik toleransları



Şekil 3-5 Yuvarlak Kapak Plakasının geometrik toleransları

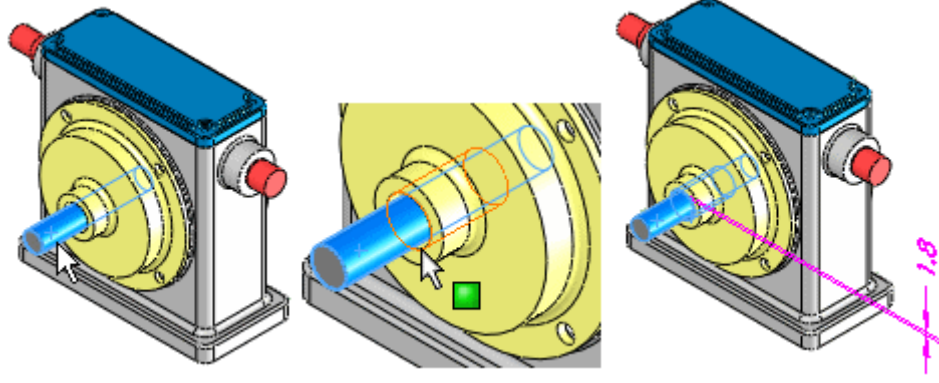


Şekil 3-6 Sonsuz Dişli Şaftın geometrik toleransları

### 4.3. TolAnalyst Yardımıyla Çözüm

Bu bölümde WC ve RSS yöntemlerinin sonuçlarını elde etmek için Solidworks içerisinde eklenti olarak gelen TolAnalyst programı kullanılacaktır. Bu araç, çok karmaşık problemlerde gerekli olan kritik ölçüyü etkileyen tolerans zincirini kendisi oluşturarak birçok hesaplamayı gömülü olarak yapmaktadır. Böylece kritik tolerans analizinde en karışık ve yorucu olan aşamalar, tolerans yığılmasındaki elemanların tespiti ve bunların hassasiyetini hesaplama aşaması atlanmış olunacaktır. Çünkü önerilen yöntemde sadece girdi ve çıktı değerleriyle ilgilenilmektedir. Dolayısıyla MCS ve bu çalışmada incelenen vekil model tabanlı çözümler için iyi bir karşılaştırma mümkün olacaktır. Kabaca bu bölümde şu işlemlerin detayları anlatılacaktır: iki unsur arasın bizim için önemli olan kritik ölçünün tanımlanması, bileşenlerin montaj sırasının belirlenmesi, parçaların sınırlandırılması ve son olarak sonuçların değerlendirilmesi.

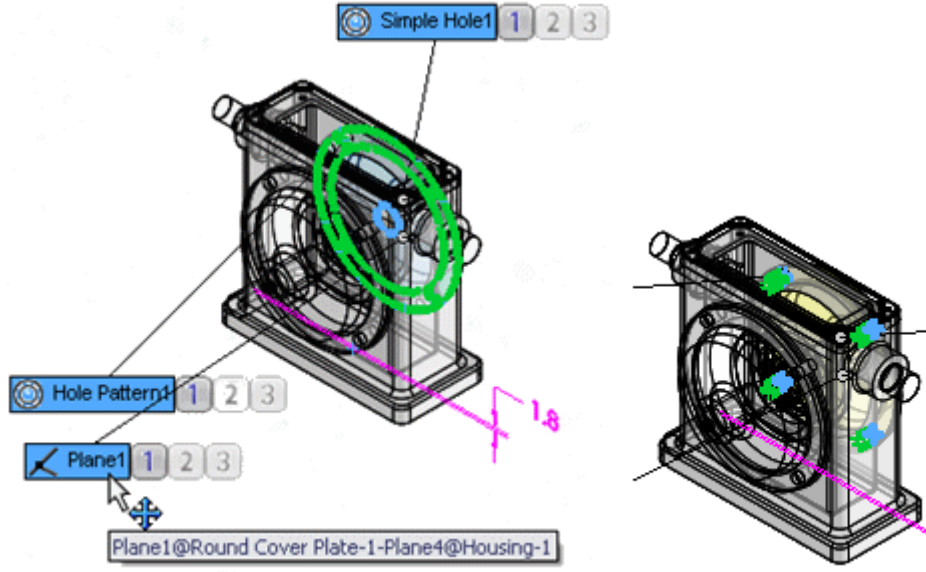
Öncelikle Solidworks öğreticileri içinde bulunan “*GearBox.sldasm*” montaj dosyası açılır (<yüklenen\_yer>\samples\tutorial\TolAnalyst\Introduction\). Kullanılan montaj elemanı, daha öncesinde DimExpert yardımıyla ölçülendirilmiş ve toleransları belirtilmiştir. Bu nedenle modelin hazırlanmasına gerek yoktur. Verilen toleranslar önceki bölümde problem tanımında gösterilmiştir. DimExpert araç kutusu içinde bulunan TolAnalyst etüdüne tıklanarak bize işlemler için yol gösterecek olan sihirbaz etkinleştirilir. Ardından “*Buradan Ölç*” araç kutusu seçilerek ilk olarak Yükseklik2@Sonsuz Dişli Şaftı-1 ardından ikinci yüzey, Basit Delik1@Yuvarlak Kapak Plakası-2 olarak ifade edilen yuvarlak kapak plakasının delik iç yüzeyi ile sonsuz dişli şaftın dış yüzeyi seçilir ve arasındaki uzaklık 1,8 mm olarak görülür. Burada iki yüzey arasındaki mesafenin “En Küçük” seçilerek ölçülmesi gerekir. Burada ölçüm ayrıca yüzeyin normali olan “N” olarak seçilmiştir silindir olduğundan çap doğrultusundadır. Seçim adımları şekilde sırasıyla Şekil 3-7’de gösterilmiştir.



Şekil 3-7 Kritik ölçü için unsurların sıralı seçimi

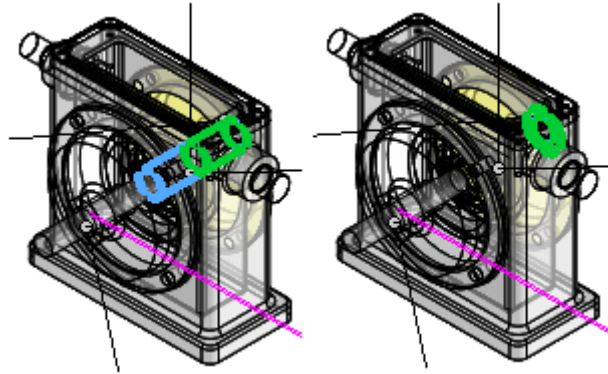
İleri tıklanarak bir sonraki tolerans sıralama adımına geçilmiştir. Bu aşamada montaj sırası ürün ağacından şu sıralamada seçilmiştir: 1. Muhafaza-1, 2. Yuvarlak Kapak Plakası-1, 3. Sonsuz Dişli Şaftı-1, 4. Yuvarlak Kapak Plakası-2. Ardından ileri o işaretine tıklanarak bir sonraki aşamaya geçilmiştir.

Sonraki adım olan montaj sınırlandırma önemli adımlardan biridir. Bu aşama model montajı yapılırken kullanılan kısıtlardan faydalanır. Bu kısıtlar montajın fonksiyon olarak nasıl çalıştığını ve bileşenler arasındaki ilişkiyi ifade ettiğinden, tolerans zinciri oluşturmada önemlidir. Bu çalışmada kullanılan eklenti çakışık, eş merkezli, uzaklık ve teğet gibi sınırlandırmalar ile çoğaltmaları kullanılabilir. Bu aşamada var olan kısıtlar arasından montaj sırasına göre 3 adet sınırlandırma sıralanır. Seçeneklerin üzerine gelindiğinde bileşenler arasındaki kısıtın ifadesi model üzerinde yanar. Burada Şekil 3-8'de gösterilen sırayla şu şekilde seçilmiştir; önce kapak plakası ile muhafaza yüzeyi üzerindeki çakışık sınırlandırmayı gösteren Düzlem1 ifadesini 1 olarak, seçilmiştir. Ardından deliklerin eşmerkezli olarak hizalanmasını sağlayan Delik Çoğaltma1 ifadesini 2 olarak seçilmiştir. Son olarak da sonsuz dişli şaft1 ile yuvarlak kapak plakası-1 arasındaki eşmerkezli delik sınırlandırmasını gösteren basit delik1 ifadesini 3 olarak otomatik seçilmiştir.



Şekil 3-8 Yuvarlak Kapak Plakası-1 için montaj sınırlandırmaları sıralı gösterimi

Ardından diğer montaj elmanı için benzer işlemler tekrarlanmıştır. İkinci olarak tolerans montajı altında, bileşenler, sonsuz dişli şaftı-1 ögesi seçilir. Şekil 3-9'de gösterildiği gibi önce Yükseklik1 ardından Düzelm2 ögesi seçilerek sonsuz dişli şaft için sınırlandırmalar tamamlanarak yuvarlak kapak montajı sınırlandırmalarına geçilmiştir.

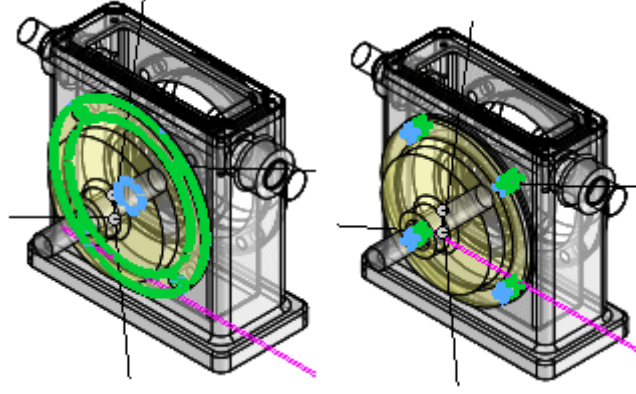


Şekil 3-9 Sonsuz Dişli Şaft-1 için montaj sınırlandırmaları sıralı gösterimi

Son olarak Kapak plakası ile muhafaza arasındaki çakışık sınırlandırma amacıyla Düzelm1 ögesi 1 olarak seçilmiştir. Ardından Kapak plakasının delik çoğaltması ve muhafazanın delik çoğaltması arasındaki eşmerkezli sınırlandırmayı da yapmak için



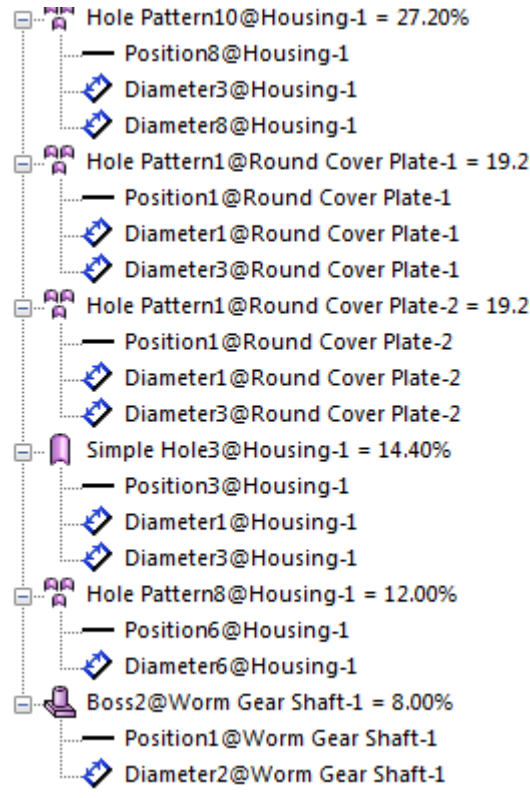
Delik ođaltma1 gesi 2 olarak seilmiřtir. İlgili sınırlandırmalar sırasıyla Őekil 3-10'de gsterilmiřtir.



Őekil 3-10 Yuvarlak Kapak Plakası-2 iin montaj sınırlandırmaları sıralı gsterimi

Biten ařamalardan sonra ileriye tıklandıđında Analiz Sonuları blmne ulařılır. Sonuların parametrelerinde her hangi bir deđiřiklik yapılmadan duyarlılık binde bařmađında alınmıřtır. Elde edilen WC ve RSS sonularının dıřında ‘‘Analiz Veri Grnm’’ blmnde analize katkı yapan her bir elemanın en az/ok yaptıđı katkı ve ilgili boyutsal ve geometrik toleranslar gsterilmiřtir. Bu problem iin elde edilen katkılar Őekil 3-11'de gsterilmiřtir.

Toleransa katkı yapan elemanlarının bilinmesi toleransların dzenlenemesi ve eniyilemeyapılabilmesi asında tasarımcı iin ok nemlidir. Bu alıřmada MCS yntemini uygulayabilmek iin bu katkı yapan toleranslardan da faydalanılmıřtır. Bu nedenle elde edilen verilerin hem sonuların karřılařtırılması hem de alıřmada nerilen yntemlerin uygulanması asından ok nemlidir.



Şekil 3-11 Dişli Kutusu Montajına katkı sağlayan eleman ve toleransların listesi ile en yüksek duruma göre yüzde katkıları

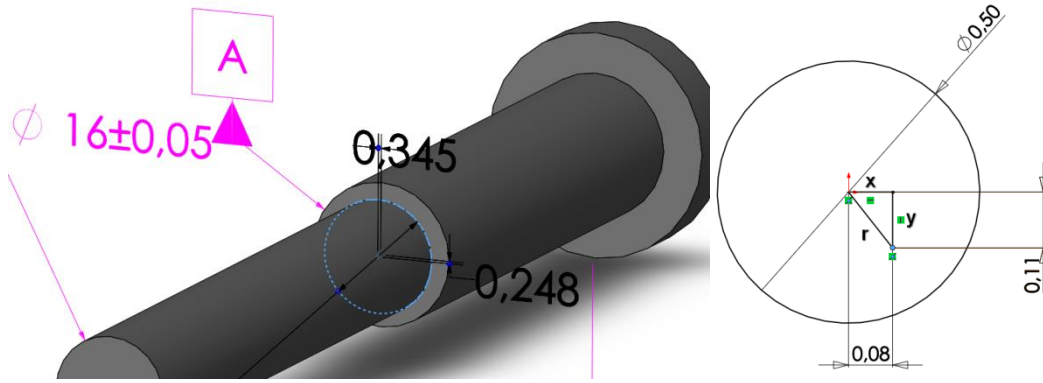
Solidworks TolAnalyst eklentisi, analiz yöntemi olarak MCS kullanamamaktadır. Fakat bu çalışmada incelenilen yöntemler arasında MCS vardır. Bunun yanından vekil model yöntemiyle yapacağımız analizin modeli oluşturulurken mutlaka deneysel ya da istatistiksel olarak girdi ve çıktılar kullanılmaktadır. Önceki örnekte bu veriler için girdi olarak rastsal değişken üretmiş ve tasarım fonksiyonuna uygulayarak çıktıları kullanılmıştı. Bu problem üç boyutlu ve daha karmaşık olduğundan tasarım fonksiyonu oluşturulmamış dolayısıyla sistemin cevapları da yoktur. Dolayısıyla bu problemde yine TolAnalyst'den elde edilen katkı yapan toleransların değerleri kullanarak rastsal değişken olarak üretilmiştir. Ardından Solidworks yazılımının değişen ölçülere karşılık güncellenen montaj ölçüleri de kullanarak kritik ölçünün değerleri yani sistemin yanıtı elde edilebilir. Fakat onlarca kez her bir girdinin el ile değiştirilmesi ve yanıtının alınması tahmin

edelebileceği gibi büyük bir zaman kaybı olduğunda bu problem de yazılımın API desteği kullanılarak bahsedilen işlemlerin yapılması Macro yazılarak otomatikleştirilmiştir.

MCS yöntemi uygulanırken temel olarak ilgili toleransa bağlı olarak her bir katkı yapan ölçü değiştirilir ve montaj tekrar kurularak kritik boşluk programın ölçme özelliği kullanılarak ölçülür. Fakat bu yöntem uygulanırken Soldiworks Öğreticiler içinde bulunan model olduğu gibi kullanılmamıştır. Çünkü yapılan bazı değişiklikler bazı sınırlandırmalar nedeniyle doğru sonuç vermemiştir. Bunun için standart montaj içinde iki montaj ilişkisi değiştirilmiştir. İlk olarak Muhafaza parçasının Yuvarlak Kapak Plakası ile kapandığı yuvarlak delik Yuvarlak Kapak Plakasının şaft deliği ile eş merkezli olarak tanımlanmıştır. Bu MCS uygulandığında değişen ölçüler nedeniyle her zaman eş merkezli olamayacağı “Concentric1 (Housing<1>Round Cover Plate <1>)” olarak ifade edilen birleştirme pasifleştirilmiştir. İkinci olarak ise arkada kullanılan Yuvarlak Kapak Plakası nın şaft deliği ile önden montajlanan Sonsuz Dişli Şaftı eş merkezli olarak tanımlanmış. Fakat bu da ölçüler değiştiğinde her zaman mümkün olmayacaktır. Bu nedenle “Concentric6 [Worm Gear Shaft<1>, Round Cover Plate <2>)” olarak ifade edilen montaj ilişkisi pasifleştirilmiştir. Aksi halde değiştirilen ölçünün ardından tekrar kurulan montaj hata vermektedir.

Standart olarak verilen montaj ilişkileri üzerinde yapılan değişikliklerin dışında toleransları değiştirmek için bazı düzeltmeler de yapılmıştır. Şekil 3-11’de görüldüğü gibi sadece doğrusal değil, konum ve pozisyon kontrolü sağlayan toleranslar da tolerans analizine etki etmektedir. Boyutsal toleranstaki değişikliği model oluşturulurken kullanılan tanımların değiştirilmesi ile mümkün olmuştur. Fakat pozisyon değişiklikleri için ilgili datuma göre iki eksenle tanımlanacak şekilde model ölçülendirilmiş ve bunlar daha sonra kod ile tolerans değerlerine uygun olarak değiştirilmiştir. Örneğin Şekil 3-6’da gösterilen sonsuz dişli şaftın toleranslarında daralmış çapa sahip olan şaftın bir kısmı için A referans yüzeyine göre merkezin 0,5mm çaplı bir çemberin içinde kalacak şekilde her hangi bir merkezden geçmesi kabul edilebilir olarak belirlenmiştir. Dolayısıyla Şekil 3-12’de gösterildiği gibi model üzerinde tanımlanan şaftın uç kısmının merkezi  $r$  değerinin ucu olarak  $x$  ve  $y$

doğrusal ölçüsü ile tanımlanmıştır. Daha sonra  $r$  değerinin 0,5 mm den düşük kalması için öncelikler  $x$  değeri 0,5 den düşük olacak şekilde rastgele üretilmiş ve  $y$  değeri  $y^2=0,5^2-x^2$  formulu ile  $y$  belirlenmiş ve bu da belirlenen  $y$  den küçük olacak rastgele değer üretilmiştir. Daha sonra  $x$  ve  $y$  rastgele değişkeni model üzerinde değiştirilmiştir.



Şekil 3-12 Pozisyon tolerans değişimi için uygulanan yöntemin gösterimi

Montaj ve parça modeli üzerinde yapılan bu tanımlama ve düzeltmelerin ardından rastgele değişkenler üretilip model üzerine uygulanmıştır. Değişkenlerin rastgele değerlerini üretmek için MATLAB yazılımında çalışan EK 9’de gösterilen “gbox\_rand.m” fonksiyonu yazılmıştır. Bu fonksiyon Şekil 3-11’ de gösterilen tüm katkı yapan toleranslar için rastsal değer üretir ve “gbox.xlsx” dosyasına kaydeder. Bunun için EK 10’da verilen “gbox\_macro.swp” macro dosyası ile işlem gerçekleştirilmiştir. Bu macro dosyası dört modülden oluşur. "Main\_Module" asıl çalıştırılması gereken modül olup, asıl olarak model üzerinde değişiklikleri yapıp montajı yeniden kurmaya yarar. Bu modül diğer üç modülü kullanarak bunu yapar. "Get\_Excel" isimli modül, excel üzerine kaydedilen, MATLAB yazılımı aracılığı ile üretilen değişkenleri dbIData isimli dizi üzerine kaydeder. "Measure\_Faces" isimli modül ise montaj dosyası üzerinde belirtilen yüzeyleri seçerek yazılımın "Ölçüm" aracını kullanarak istenilen ölçümü gerçekleştirerek değerleri bir değişkene atar. "Save\_Excel" modülü ise her bir değişim kümesini değiştirdikten sonra, ölçüm

değerlerini aynı excel dosyası üzerine kaydeder. MCS uygulaması için "gbox\_rand.m" çalıştırılarak değişkenleri oluşturmak ve ardından Solidworks Macro editörü yardımıyla "gbox\_macro.swp" macro "Main\_Module" içerisindeki "Location" parametresi macronun kaydedildiği klasör ile aynı olacak şekilde dosya yolunu değiştirmek ve "main" modulu koşturmaktır.

#### **4.4. Vekil Model Aracılığıyla Çözüm**

Yanıt yüzey yöntemi için önceki problemde olduğu gibi MATLAB içerisinde gömülü olarak gelen fonksiyonlardan faydalanılmıştır. Bunun yanında sistemin girdi ve yanıtları için macro kullanarak elde edilen sonuçlar kullanılmıştır. Fakat bu sonuçlardan yaklaşık %2'si ölçüm ya da katıların üst üste binmesinden kaynaklandığından bu yöntemde uygulanmak üzere veri setine dâhil edilmemiştir. Bu nedenle 975 örneklem üzerinden çözüm yapılmıştır. Tüm bu hesaplamalar ekte verilen "gbox\_rsm.m" fonksiyon dosyası ile gerçekleştirilmiştir. Bu yöntem uygulanırken model oluşturmak amacıyla 16 girdi değişkenine karşılık 160 ve 320 olmak üzere iki tip önermeklem adedi kullanılmıştır. Her bir model için; doğrusal, etkileşim, karesel ve önkaresel modelleri kullanılmıştır.

Kriging yöntemi önceki problemde olduğu gibi MATLAB üzerinde çalışan DACE araç kutusu yardımıyla yapılmıştır. Genel olarak özetlemek gerekirse MATLAB üzerinde çalışan ve her bir değişken için rastgele sayılar üreten EK 9'da verilen fonksiyon yardımıyla veri seti oluşturulur ve "gbox.xlsx" üzerine kaydedilir. Önceki problemde MCS yöntemini uygulamak için yazılan Macro, Solidworks yazılımı üzerinde çalıştırdığımızda "gbox.xlsx" üzerindeki rastsal değişkenleri okuyarak modele uygular ve hemen yanına merkezden merkeze, en küçük ve en büyük mesafeleri ölçerek kaydeder. Diğer sütün ise en küçük mesafe nin tam tersi tarafındaki diğer boşluğu ifade eder. Burada iki silindir içindeki ekzantrik olarak konumlanmış diğer silindir ile arasındaki boşluğun ölçülmeye çalışıldığına dikkat edilmelidir. Bu neden rastgele ölçümlerde macro yardımıyla sadece en küçük değerleri kaydettiğimizden karşısındaki değer de hesaplanarak tabloya kayıt edilir.

Bu ölçümler içinde rastgele olarak daha sonra sistemin yanıtları olarak değerlendirilecektir. EK 11’de verilen “gbox.km.m” fonksiyonu MATLAB üzerinde çalıştırılarak Km yöntemini uygular ve aynı excel dosyası üzerine yeni bir sayfa açarak ilgili bilgileri yazdırır. KM ile elde edilen sonuçlar RSM ile benzer şekilde 160 ve 320 örneklemlilik olmak üzere iki tip model kullanılmıştır. Bu model benzetim için kullanılmak üzere RSM ile aynı 975 rassal değişken kullanılarak hesaplanmıştır.

#### 4.5. Bulgular

TolAnalyst ile uygulanan çözümde elde edilen WC ve RSS sonuçları Çizelge 3-5’de gösterilmiştir. Normal dağılımlı varsayılan RSS’e göre elde edilen sonuçların ortalaması alınıp üç noktaya olan uzaklığı, standart sapması 0,825 olarak hesaplanmıştır.

Çizelge 3-5 TolAnalyst ile elde edilen sonuçlar

Nominal	1.8
WC En Küçük	-1.35
WC En Büyük	5
RSS En Küçük	0.982
RSS En Büyük	2.632
<i>RSS Standart Sapma</i>	<i>0.825</i>

MCS yönteminde elde edilen sonuçlar değerlendirilirken bazı hatalarla karşılaşmış ve bunlar elenerek değerlendirilmiştir. Yüzey seçim yönteminden kaynaklanan ya da toleransların gereği ölçümlerin alınamadığı durumlarla, ya da katıların üst üste gelmesinden dolayısıyla oluşan ölçüm problemleriyle karşılaşmıştır. Hatalar sıralandığında; merkezden en küçük ve en büyük mesafe ölçümlerinin aynı olduğu farkedilmektedir. Bu hatalar alınan excel verileri üzerindeki son sütunda notlandırıldığı gibi “geçersiz seçim kombinasyonu” ya da “kesişim var” şeklinde belirtilen hatalı satırlardır. Genelde 1000 ölçüm için yaklaşık 25 adet bu tip hatayla karşılaşmıştır. Bu değerler montajlanamaz olduğundan hurda kabul edilebilir. Bu değerler analizler uygulanmadan önce silinmiştir. Çizelge 3-6’da MCS ile elde edilen sonuçlar verilmiştir.

Çizelge 3-6 Dişli Kutusu Montajı için MCS sonuçları

Kontrollü Montaj Değişkenlerinin Çıktıları / Benzetim Sayısı	100	1000
Ortalama [radyan]	1.73883	1,8104
Standart Sapma [radyan]	0.6863	0,7845
Skewness	0.0616	-0,0185
Kurtosis	1.9792	2.0195
En Küçük	1.0524	1.0259
En Yüksek	2.4251	2.5949

Çizelge 3-6’da görüldüğü üzere aslında istatistiksel olarak yeterli adet kabul edilmeyen 100 adet benzetim için sonuçlar ile 1000 adet örneklem için sonuçlar alınmıştır. Dişli Kutusu Montajına uygulana TolAnalyst ile elde edilen Çizelge 3-5’deki RSS sonuçları ile Çizelge 3-6’da 1000 örneklemlerle MCS ile elde edilen sonuçlar karşılaştırıldığında ortalama değerlerinin ve standart sapmalarının beklendiği gibi yakın çıktığını görülmektedir. Burada ortalama değer için yaklaşık %1 hata, standart sapma için ise yaklaşık %5’lik hata gözlenmiştir.

RSM ile elde edilen sonuçlar Çizelge 3-7’de verilmiştir. RSM için alınan sonuçlar ile Çizelge 3-6’ MCS için alınan 1000 örneklemlerle sonuçlar karşılaştırıldığında; etkileşim modelinin hem ortalama değer hem de standart sapma için en yakın sonucu verdiği görülmektedir (önceki örnek problemde ise karesel model en yakın sonucu vermişti).

Çizelge 3-7 Dişli Kutusu Montajı için RSM ile elde edilen sonuçlar

Örneklem adeti	160	160	160	160	320	320	320	320
Regresyon Tipi	doğrusal	etkileşim	karesel	önkaresel	doğrusal	etkileşim	karesel	önkaresel
Ortalama [radian]	1,6530	1,8799	1,2486	0,7974	1,6381	1,8901	2,3985	1,9248
Standart Sapma [radian]	0,2080	2,3057	1,7556	0,9671	0,2094	0,9075	1,0609	0,3141
Skewness	0,0263	-0,1623	-0,4054	-1,2008	-0,0150	-0,1322	1,2150	0,5790
Kurtosis	2,9457	6,5081	5,6592	5,2574	3,4199	9,9924	6,5727	5,1852
En Küçük	1,8610	4,1856	3,0042	1,7645	1,8475	2,7975	3,4593	2,2390
En Yüksek	1,4449	-0,4258	-0,5069	-0,1697	1,4287	0,9826	1,3376	1,6107
Hesaplama Zaman [saniye]	1,5821	1,1750	1,2484	1,1246	1,5257	1,2723	1,3519	1,3636
Tahmin	975	975	975	975	975	975	975	975

KM ile elde edilen sonuçlar Çizelge 3-8’de gösterilmiş olup Çizelge 3-7’de verilen RSM sonuçları ile karşılaştırılabilir. KM sonuçları Çizelge 3-6’da verilen MCS sonuçlarıyla karşılaştırıldığında; 160 örneklemlile modelle yapılan vekil model tahminlerinde sabit (*regpoly0*) ve doğrusal (*regpoly1*) eğilim modelleriyle elde edilen sonucun, MCS ile elde edilen ortalama değerlerine oldukça yakın olduğu ancak standart spama tahmininde yetersiz olduğu gözlenmiştir. 320 örneklem kullanıldığında ise hem ortalama hem de standart sapma tahminlerinin sonuca yakınsadığı görülmektedir. Fakat karesel eğilim modeli (*regpoly2*) kullanılarak oluşturulan KM sonuçlarının hatasının oldukça yüksek olduğu görülmektedir.

Çizelge 3-8 Dişli Kutusu Montajı için KM ile elde edilen sonuçlar

Örnekleme adedi	160	160	160	320	320	320
Regresyon Tipi	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2
Ortalama [radian]	1,8213	1,8429	3,7978	1,9094	1,9094	1,9401
Standart Sapma [radian]	0,3016	0,3631	3,3303	0,4708	0,4708	0,8122
Skewness	0,0093	-0,081	1,3766	-0,6826	-0,6826	0,6785
Kurtosis	13,193	5,8893	6,1903	6,0082	6,0082	6,7283
En Küçük	2,1229	2,2060	7,1281	2,3803	2,3803	2,7524
En Yüksek	1,5196	1,4797	0,4675	1,4386	1,4386	1,1278
Hesaplama Zamanı [saniye]	1,1630	1,2070	1,3751	1,3288	1,3288	1,3003
Tahmin	975	975	975	975	975	975



## 5. DEĞERLENDİRME

Bu bölüme elde edilen sonuçlar gözden geçirilecek ve bulgular değerlendirilmiştir. Bunun yanında literatüre yapılan katkılar, çalışmadaki sınırlar ve ileriye dönük çalışmalar için öneriler ifade edilecektir.

### 5.2. Sonuçlar ve Tartışma

Bu çalışmada, günümüz endüstrisinde kalite ve maliyet gibi önemli konuların iyileştirilmesi için gerekli en önemli unsurlardan olan toleransların en iyi şekilde nasıl analiz edileceği araştırılmıştır. Genel olarak kabul görmüş ve güvenilir olan geleneksel yöntemler araştırılmıştır. Günden güne artan bilgisayar gücüne rağmen hala karmaşık ve maliyetli olan analiz yöntemleri yerine, son yıllarda sıkça kullanılan vekil modellerin tolerans analizinde kullanılması ele alınmıştır. Bu yöntemler iki boyutlu doğrusal olmayan tasarım fonksiyonuna sahip bir problem ve daha karmaşık üç boyutlu problem üzerinde uygulanmıştır.

İki boyutlu problemde (Fortini'nin kavrama montajı probleminde), geleneksel çözüm WC ve RSS yöntemi uygulanmıştır. Normal dağılımlı olduğu varsayılan RSS ile elde edilen sonuca göre MCS yöntemi ile elde edilen sonuca bakıldığında tutarlı olduğu görülmektedir. Ardından MCS yöntemi uygulanarak dört değişkenli bu problem için 10,000 örneklemlerle benzetim tatmin eden sonuç verdiği gibi 100,000 örneklemlerle sonucun yüksek hassasiyetle yakınsadığı görülmüştür. Bu yöntem vekil model yöntemi için de referans kabul edilmiştir. RSM ve KM tabanlı tolerans analizi ile yüksek hassasiyette sonuç tahmini yapılabilmektedir. Genellemek gerekirse değişken sayısının on katı kadar örneklem adeti ile yakınsayan sonuç elde edilebilmiştir. Bu problem için RSM ve KM modelleri yakın sonuçlar vermiştir.

Üç boyutlu bir örnek problem olarak kullanılan bir dişli kutusu montajı için, TolAnalyst isimli eklenti paketi katı model üzerinden parametreleri seçilerek hesaplanmıştır. Elde edilen WC ve RSS sonuçları doğru kabul edilip referans kabul edilmiştir. Bahsi geçen eklenti MCS uygulaması gerçekleştirilemediğinden

Solidworks yazılımını üzerinde otomatik olarak çalışan bir macro yazılmıştır. Bu macro ile TolAnalyst yardımıyla incelenmekte olan kritik ölçüyü etkileyen toleranslar tespit edilmiş ve bunlar elle değiştirilerek model her seferinde tekrar kurulup kritik ölçüdeki etkisi incelenmiştir. Elde edilen sonuçlar 1000 örneklem için geleneksel çözümle elde edilen sonuçla örtüşmektedir. Bu da macronun doğru çalıştığını ve yöntemlerin tutarlı olduğunu göstermiştir. Burada bazı ölçüm hataları ile karşılaşmış ve bunlar benzetim sonuçlarının değerlendirilmesinde kullanılmamıştır. Macro ile yapılan işlemler sonucunda elde edilen girdi ve yanıtlar vekil model tabanlı tolerans analizi yöntemlerinde kullanılmıştır.

Üç boyutlu bir örnek problem için RSM ile elde edilen sonuçlar incelendiğinde, etkileşim modelinin en doğru sonuç verdiği gözlenmiştir. KM ile elde edilen sonuçlar incelendiğinde, sabit veya doğrusal eğilim modelinin kullanımının uygun olduğu, ancak karesel eğilim modelinin kullanımının bu örnek problem için uygun olmadığı gözlenmiştir. RMS ile KM karşılaştırıldığında ise, KM'nin RSM'den daha doğru sonuçlar verdiği gözlenmiştir.

Bu çalışma MCS yönteminin popüler katı modelleme programı üzerinde otomatikleştirilerek uygulanması açısından önemli katkılar sunmuştur. Bunun dışına özellikle üç boyutlu model üzerinde daha önce tolerans analiz yöntemi olarak uygulanmamış RSM ve KM vekil modelleriyle analiz edilmiştir. Ek olarak bunların sonuçlarının geleneksel yöntemlere karşılaştırılması açısından literature katkı sağlamıştır.

### 5.3. İleriye Yönelik Çalışmalar

Bu çalışmada, imalat sanayinde kullanılan mekanik toleransların analizinin vekil model tabanlı matematiksel teknikler kullanılarak, geleneksel yöntemlere göre verimlilik ve maliyet açısından iyileştirilmesi değerlendirilmiştir. Bu çalışma WC, RSS, MCS geleneksel yöntemleri, RSM ve KM vekil tabanlı modeller ve örneklerde kullanılan geometrik toleranslar ile sınırlıdır.

Bu çalışma aşağıda sıralanan çalışmalarla genişletilebilir:

- Geleneksel tolerans analiz yöntemleri dışında geliştirilmiş diğer yöntemler ve diğer geometrik toleranslarla birlikte çözümün karşılaştırması,
- Diğer vekil model tiplerinin veya vekil model bütününün kullanılması,
- Anlaşılır ve kullanılabilirlik geliştirilmesi açısından kodların birleştirilip genelleştirilmiş ve kullanıcı arayüzü oluşturulması,
- Maliyet ve kalite teorileri ile bütünleştirilerek tolerans analizi yapılması.

## 6. EKLER

### EK 1 rss\_mcs\_clutch.m fonksiyon kodları

```
                                rss_mcs_clutch.m
function rss_mcs_clutch( ~ )

clear; clc;          % Clean cache
tic;                % Time Counter Start

% Design Function (y)
syms x1x2x3x4
y=acos((x1+((x2+x3)./2))./(x4-((x2+x3)./2)));
yrad=subs(y, [x1 x2 x3 x4], [55.29 22.86 22.86 101.59]);
yrad=double(yrad)
ydeg=radtodeg(yrad);

% Sensitivity Calculation (a)
dydx1=diff(y,x1);
dx1=subs(dydx1, [x1 x2 x3 x4], [55.29 22.86 22.86 101.59]);
dx1=double(dx1);

dydx2=diff(y,x2);
dx2=subs(dydx2, [x1 x2 x3 x4], [55.29 22.86 22.86 101.59]);
dx2=double(dx2);

dydx3=diff(y,x3);
dx3=subs(dydx3, [x1 x2 x3 x4], [55.29 22.86 22.86 101.59]);
dx3=double(dx3);

dydx4=diff(y,x4);
dx4=subs(dydx4, [x1 x2 x3 x4], [55.29 22.86 22.86 101.59]);
dx4=double(dx4);

% Analysis
% Nominals
x1=55.29; x2=22.86; x3=22.86;x4=101.69;
% Tolerances
tx1=0.1016; tx2=0.01016; tx3=0.01016; tx4=0.2032;
%RSS
rss=sqrt((dx1*tx1)^2+(dx2*tx2)^2+(dx3*tx3)^2+(dx4*tx4)^2)
rss_deg=radtodeg(rss);
%WC
wc=abs(dx1*tx1)+abs(dx2*tx2)+abs(dx3*tx3)+abs(dx4*tx4)
wc_deg=radtodeg(wc);

toc %Time Counter End
end
```

## EK 2 mcs\_clutch.m fonksiyon kodları

```

                                mcs_clutch.m
function mcs_clutch( ~ )

clear; clc;           % Clean cache
tic;                 % Time Counter Start
% Design Specification
mu_g=0.122;         std_g=0.035;
USL=mu_g+std_g;
LSL=mu_g-std_g;
% Design Function (y) Parameters
mu_x1=055.29;      std_x1=0.1016;
mu_x2=022.86;      std_x2=0.01016;
mu_x3=022.86;      std_x3=0.01016;
mu_x4=101.69;      std_x4=0.2032;
% Monte Carlo Simulation Parameters
N=100000;          % Simulation Number
x1=icdf('Normal',rand(1,N),mu_x1,std_x1);
x2=icdf('Normal',rand(1,N),mu_x2,std_x2);
x3=icdf('Normal',rand(1,N),mu_x3,std_x3);
x4=icdf('Normal',rand(1,N),mu_x4,std_x4);
%Design Function (y)
y=(x1+((x2+x3)./2))./(x4-(x2+x3)./2);
y=acos(y); y=real(y);
%Outputs
mean_y=mean(y)      % 1.Moment Mean
std_y=std(y)        % 2.Moment Standard Deviation
skw_y=skewness(y)  % 3.Moment Skewness
krt_y=kurtosis(y)  % 4.Moment Kurtosis
% Scrap Calculations
U=y>USL; upper_reject_count=sum(U);
prc_upper=upper_reject_count/N*100
L=y<LSL; lower_reject_count=sum(L);
prc_lower=lower_reject_count/N*100
maximum=max(y)
tot_prc_reject=prc_upper+prc_lower
%Histogram
figure,
hist(y,25)
hold on ,
plot([LSL,LSL],[0,18000],'Color','r');plot([USL,USL],[0,18000]
,'Color','r');
plot([mu_g,mu_g],[0,18000],'Color','c');
title('Histogram of design function distribution');
xlabel('Occurrences');
ylabel('Number of sequence reads/Frequency');

toc           % Time Counter End
end

```

### EK 3 mcs\_vekil\_clutch.m fonksiyon kodları

```
                                mcs_vekil_clutch.m
function [s, y] = mcs_vekil_clutch( Npredict )

% Variables
mu_x1=055.29;    std_x1=0.1016;
mu_x2=022.86;    std_x2=0.01016;
mu_x3=022.86;    std_x3=0.01016;
mu_x4=101.69;    std_x4=0.2032;

% Monte Carlo Simulation
% N % Simulation Number
x1=icdf('Normal',rand(1,Npredict),mu_x1,std_x1);
x2=icdf('Normal',rand(1,Npredict),mu_x2,std_x2);
x3=icdf('Normal',rand(1,Npredict),mu_x3,std_x3);
x4=icdf('Normal',rand(1,Npredict),mu_x4,std_x4);
s=[x1',x2',x3',x4'];

% Design Function
y=(x1+((x2+x3)./2))./(x4-(x2+x3)./2);
y=acos(y); % y=real(y);

toc      % Time Counter End
end
```

## EK 4 rsm\_clutch.m fonksiyon kodları

```

                                rsm_clutch.m
function rsm_clutch( ~ )

clear; clc;           % Clean cache
tic;                 % Time Counter Start
filename='C:\Users\Utku\Documents\thesis\3_thesis\ekler\clutch
.xlsx';
sheet='RSMfixed';
firstRow = 23;
firstCol = 13;
%model='linear'; % Constant and linear terms (the default)
%model='interaction'; % Constant, linear, and interaction
terms
%model='quadratic'; % Constant, linear, interaction, and
squared terms
model='purequadratic'; % Constant, linear, and squared terms
Nmodel=100;
Npredict=100000;
% Regression Model
%[s,y] = MCS(Nmodel); % Random
% From storage
    load('sy100k.mat')
s=s(1:Nmodel,1:end); % what you need
y=y(1:Nmodel,1); % what you need
    stats = regstats(y, s, model);
    b = stats.beta; % Model coefficients
% Prediction
    [s,~]=mcs_vekil_clutch(Npredict);
% Sabitten çek
    load('sy100k.mat')
s=s(1:Npredict,1:end); % what you need
    S = x2fx(s, model);
    YY = S*b; % Prediction of S
    [mean_y, std_y, skw_y, krt_y, prc_upper, prc_lower,
tot_prc_reject] = statistic(YY, Npredict);
time=toc;
% cellRange =
[firstCol,num2str(firstRow),':',lastCol,num2str(lastRow)]
firstColletter = xlcolumnletter(firstCol);
cellRange = [firstColletter,num2str(firstRow)];
xlswrite(filename,Nmodel,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,{model},sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,mean_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,std_y,sheet,cellRange)

```

```

firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,skw_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,krt_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,prc_upper,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,prc_lower,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,tot_prc_reject,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,time,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,Npredict,sheet,cellRange)

winopen(filename);

toc      % Time Counter End
end

```



## EK 5 statistic.m fonksiyon dosyası

```
                                statistic.m
function [mean_y, std_y, skw_y, krt_y, prc_upper, prc_lower,
tot_prc_reject] = statistic(y,N)
% Outputs
mean_y=mean(y);                % 1.Moment Mean
std_y=std(y);                  % 2.Moment Standard Deviation
skw_y=skewness(y);            % 3.Moment Skewness
krt_y=kurtosis(y);            % 4.Moment Kurtosis
% Rejections
mu_g=0.122;                    std_g=0.035;
USL=mu_g+std_g;
LSL=mu_g-std_g;

U=y>USL;  upper_reject_count=sum(U);
prc_upper=upper_reject_count/N*100;

L=y<LSL;  lower_reject_count=sum(L);
prc_lower=lower_reject_count/N*100;

tot_prc_reject=prc_upper+prc_lower;
end
```

## EK 6 krig\_clutch.m fonksiyon dosyası

```

                                krig_clutch.m
function krig_clutch( ~ )

clear; clc;           % Clean cache
tic;                 % Time Counter Start
filename='C:\Users\Utku\Documents\thesis\3_thesis\ekler\clutch
.xlsx';
sheet='KRIGINGfixed';
firstRow = 23;
firstCol = 10;
regpol=2;
Nmodel=100;
Npredict=100000;
% Regression Model
% [s,y] = MCS(Nmodel); % Random
% Fixed
        load('sy100k.mat')
s=s(1:Nmodel,1:end); % What you need
y=y(1:Nmodel,1);    % What you need
    [dmodel,~]=dacefit(s, y, @regpoly2, @corrGauss, 2);
% Prediction
% [s,~]=MCS(Npredict); % Random
% Fixed
        load('sy100k.mat')
s=s(1:Npredict,1:end); % What you need
    YY=predictor(s, dmodel);
    [mean_y, std_y, skw_y, krt_y, prc_upper, prc_lower,
tot_prc_reject] = statistic(YY, Npredict);

time=toc;
% cellRange =
[firstCol,num2str(firstRow),':',lastCol,num2str(lastRow)]
firstColletter = xlcolumnletter(firstCol);
cellRange = [firstColletter,num2str(firstRow)];
xlswrite(filename,Nmodel,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,{regpol},sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,mean_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,std_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,skw_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,krt_y,sheet,cellRange)

```

```

firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,prc_upper,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,prc_lower,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,tot_prc_reject,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,time,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,Npredict,sheet,cellRange)

winopen(filename);

toc      % Time Counter End
end

```

## EK 7 Fortini'nin Kavrama Montajı için RSM sonuçları

Modelling	20	20	20	20	40	40	40	40	100	100	100	100
Regression Type	linear	interaction	quadratic	purequadratic	linear	interaction	quadratic	purequadratic	linear	interaction	quadratic	purequadratic
Mean [radian]	0,1279	0,1279	0,1281	0,1282	0,1286	0,1288	0,1279	0,1284	0,1281	0,1283	0,1280	0,1281
Standard Deviation	0,0247	0,0230	0,0232	0,0235	0,0247	0,0248	0,0232	0,0238	0,0233	0,0235	0,0233	0,0233
Skewness	-0,0565	0,0480	-0,6697	-0,3696	-0,0590	-0,0517	-0,6809	-0,3782	-0,0576	-0,1392	-0,5967	-0,4063
Kurtosis	2,9607	3,2636	3,5029	3,1130	2,9596	2,9694	3,5023	3,1282	2,9610	2,9874	3,3728	3,1671
Rejects at Upper Limits [%]	11,7	10,2	9,1	10,2	12,4	12,7	9	10,7	10,7	10,6	9,4	9,9
Rejects at Lower Limits [%]	4,7	3,5	5,3	4,6	4,5	4,4	5,3	4,8	3,9	4,3	5,2	5
Total Rejects [%]	16,4	13,7	14,4	14,8	16,9	17,1	14,3	15,5	14,6	14,9	14,6	14,9
Elapsed time [seconds]	0,625795747	0,24101517	0,072004362	0,074603008	0,253519633	0,086194075	0,088510493	0,088051656	0,165459852	0,103345485	0,104326873	0,098739163
Prediction	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
Modelling	20	20	20	20	40	40	40	40	100	100	100	100
Regression Type	linear	interaction	quadratic	purequadratic	linear	interaction	quadratic	purequadratic	linear	interaction	quadratic	purequadratic
Mean [radian]	0,1297	0,1295	0,1298	0,1300	0,1304	0,1307	0,1296	0,1302	0,1299	0,1300	0,1297	0,1299
Standard Deviation	0,0245	0,0229	0,0227	0,0231	0,0245	0,0247	0,0226	0,0235	0,0231	0,0232	0,0228	0,0230
Skewness	-0,0362	0,0338	-0,6716	-0,3524	-0,0375	-0,0234	-0,6838	-0,3555	-0,0368	-0,1227	-0,5918	-0,3793
Kurtosis	2,9841	3,1442	3,6672	3,1713	2,9850	3,0006	3,6737	3,1807	2,9853	3,0335	3,5087	3,2184
Rejects at Upper Limits [%]	13,25	11,28	9,93	11,42	13,78	14,2	9,5	12,24	11,87	12	10,29	11,27
Rejects at Lower Limits [%]	4,1	3,03	4,36	3,97	3,83	3,79	4,38	4,1	3,15	3,44	4,32	3,93
Total Rejects [%]	17,35	14,31	14,29	15,39	17,61	17,99	13,88	16,34	15,02	15,44	14,61	15,2
Elapsed time [seconds]	0,077888413	0,077577105	0,077073369	0,077305139	0,113319293	0,10219647	0,091541891	0,099278391	0,126610577	0,109236378	0,121261479	0,103774388
Prediction	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000
Modelling	20	20	20	20	40	40	40	40	100	100	100	100
Regression Type	linear	interaction	quadratic	purequadratic	linear	interaction	quadratic	purequadratic	linear	interaction	quadratic	purequadratic
Mean [radian]	0,1294	0,1292	0,1295	0,1296	0,1301	0,1303	0,1293	0,1299	0,1296	0,1297	0,1294	0,1296
Standard Deviation	0,0247	0,0232	0,0228	0,0233	0,0247	0,0249	0,0228	0,0237	0,0233	0,0234	0,0230	0,0232
Skewness	0,0117	0,0968	-0,6169	-0,3030	0,0115	0,0239	-0,6323	-0,3060	0,0116	-0,0821	-0,5414	-0,3300
Kurtosis	2,9758	3,1064	3,4926	3,1056	2,9763	2,9980	3,5146	3,1125	2,9761	3,0093	3,3715	3,1446
Rejects at Upper Limits [%]	13,285	11,502	9,932	11,537	13,883	14,291	9,621	12,262	11,992	12,063	10,315	11,277
Rejects at Lower Limits [%]	4,228	3,167	4,541	4,134	3,966	3,981	4,605	4,277	3,305	3,582	4,52	4,139
Total Rejects [%]	17,513	14,669	14,473	15,671	17,849	18,272	14,226	16,539	15,297	15,645	14,835	15,416
Elapsed time [seconds]	0,12332389	0,123719011	0,128415989	0,119866155	0,134972686	0,133636801	0,14866465	0,136661785	0,165221668	0,157783218	0,147883388	0,15557285
Prediction	100000	100000	100000	100000	100000	100000	100000	100000	100000	100000	100000	100000

EK 8 Fortini'nin Kavrama Montajı için KM sonuçları

Modelling	20	20	20	40	40	40	100	100	100
Regression Type	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2
Mean [radian]	0,1176	0,1279	0,1280	0,1188	0,1287	0,1279	0,1264	0,1282	0,1280
Standard Deviation	0,0068	0,0244	0,0231	0,0079	0,0246	0,0232	0,0115	0,0231	0,0233
Skewness	0,1454	-0,0952	-0,6306	-0,3276	-0,1104	-0,6819	-0,5984	-0,1948	-0,6077
Kurtosis	12,9455	3,0315	3,4518	9,5308	3,0487	3,5213	6,0047	3,0235	3,4340
Rejects at Upper Limits [%]	0,1	11,2	9,3	0,1	11,6	9,2	0,6	10,1	9,4
Rejects at Lower Limits [%]	0,5	4,8	5	0,6	4,9	5,2	0,7	4,5	5,3
Total Rejects [%]	0,6	16	14,3	0,7	16,5	14,4	1,3	14,6	14,7
Elapsed time [seconds]	0,086270192	0,132787548	0,358348104	0,079223442	0,078475107	0,079629681	0,104509466	0,08805465	0,092174768
Prediction	1000	1000	1000	1000	1000	1000	1000	1000	1000
Modelling	20	20	20	40	40	40	100	100	100
Regression Type	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2
Mean [radian]	0,1176	0,1297	0,1297	0,1189	0,1305	0,1296	0,1267	0,1299	0,1297
Standard Deviation	0,0058	0,0242	0,0225	0,0067	0,0244	0,0226	0,0097	0,0229	0,0228
Skewness	1,0538	-0,0647	-0,6290	0,2824	-0,0764	-0,6835	-0,5232	-0,1469	-0,6010
Kurtosis	9,4357	3,0548	3,6013	7,3471	3,0709	3,6900	4,9438	3,0664	3,5702
Rejects at Upper Limits [%]	0,01	12,58	9,82	0,01	13,45	9,56	0,19	11,19	10,2
Rejects at Lower Limits [%]	0,05	4,13	4,24	0,08	4,04	4,37	0,12	3,6	4,33
Total Rejects [%]	0,06	16,71	14,06	0,09	17,49	13,93	0,31	14,79	14,53
Elapsed time [seconds]	0,169423889	0,171672315	0,171426433	0,184739526	0,187543003	0,195004116	0,237187543	0,243120771	0,242404507
Prediction	10000	10000	10000	10000	10000	10000	10000	10000	10000
Modelling	20	20	20	40	40	40	100	100	100
Regression Type	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2	regpoly0	regpoly1	regpoly2
Mean [radian]	0,1176	0,1294	0,1294	0,1188	0,1302	0,1293	0,1266	0,1296	0,1294
Standard Deviation	0,0057	0,0244	0,0227	0,0066	0,0247	0,0228	0,0098	0,0231	0,0229
Skewness	1,2222	-0,0157	-0,5743	0,2414	-0,0260	-0,6320	-0,4880	-0,0974	-0,5500
Kurtosis	8,4725	3,0405	3,4380	6,8221	3,0540	3,5292	4,7172	3,0428	3,4274
Rejects at Upper Limits [%]	0,001	12,683	9,898	0,002	13,547	9,671	0,181	11,159	10,272
Rejects at Lower Limits [%]	0,02	4,283	4,418	0,072	4,202	4,609	0,125	3,741	4,551
Total Rejects [%]	0,021	16,966	14,316	0,074	17,749	14,28	0,306	14,9	14,823
Elapsed time [seconds]	1,180270042	1,111706311	1,141470371	1,284563988	1,289555168	1,288098267	1,764748292	1,686233844	1,706844271
Prediction	100000	100000	100000	100000	100000	100000	100000	100000	100000

## EK 9 gbox\_rand.m fonksiyon dosyası

```

                                gbox_rand.m
function gbox_rand( )

% Starts with i. row
clear; clc;                    % Fresh start
N=100;                        % Simulation number
usage='model';                % To change sheet
tic;
filename='C:\Users\Utku\Documents\thesis\3_thesis\ekler\gbox.x
lsx';
if strcmp(usage,'model')
    sheet='model';
elseif strcmp(usage,'predict')
    sheet='predict';
end
% Hole Pattern10@Housing-1=27.20%
% % Position8@Housing-1 Dia0,5|A
% % % Position8@Housing-1 X Component (i,1)
x1=icdf('Normal',rand(1,N),0 ,0.5);
% % % Position8@Housing-1 Y Component (i,2)
x02=sqrt(abs((0.5.^2)-(x1.^2)));
x2=icdf('Normal',rand(1,N),0 ,x02);
% % Diameter3@Housing-1 (i,3)
x3=icdf('Normal',rand(1,N),80 ,0.1);
% % Diameter8@Housing-1 (i,4)
x4=icdf('Normal',rand(1,N),6.25 ,0.25);

% Hole Pattern1@Round Cover Plate-1=19.20%
% % Position1@ Round Cover Plate-1
% % % Position1@ Round Cover Plate-1 X Component (i,5)
x5=icdf('Normal',rand(1,N),0 ,0.5);
% % % Position1@ Round Cover Plate-1 Y Component (i,6)
x06=sqrt(abs(0.5.^2-x5.^2));
x6=icdf('Normal',rand(1,N),0 ,x06);
% % Diameter1@ Round Cover Plate-1 (i,7)
x7=icdf('Normal',rand(1,N),16.1 ,0.05);
% % Diameter3@ Round Cover Plate-1 (i,8)
x8=icdf('Normal',rand(1,N),6.25 ,0.25);

% Hole Pattern1@Round Cover Plate-1=19.20%
% % Position1@ Round Cover Plate-2
% % % Position1@ Round Cover Plate-2 X Component (i,9)
x9=icdf('Normal',rand(1,N),0 ,0.5);
% % % Position1@ Round Cover Plate-2 Y Component (i,10)
x10=sqrt(0.5.^2-x9.^2);
% % Diameter1@ Round Cover Plate-2 (i,11)
x11=icdf('Normal',rand(1,N),0 ,0.5);
% % Diameter3@ Round Cover Plate-2 (i,12)
x12=icdf('Normal',rand(1,N),0 ,0.5);

```

```

% Simple Hole3@Housing-1=14.40%
% % Position3@Housing-1
% % % Position3@Housing-1 X Component (i,13)
x13=icdf('Normal',rand(1,N),0 ,0.5);
% % % Position3@Housing-1 Y Component (i,14)
x014=sqrt(abs(0.5.^2-x13.^2));
x14=icdf('Normal',rand(1,N),0 ,x014);
% % Diameter1@Housing-1 (i,15)
x15=icdf('Normal',rand(1,N),0 ,0.5);
% % Diameter3@Housing-1 (i,16)
x16=icdf('Normal',rand(1,N),0 ,0.5);

% Hole Pattern8@Housing-1=12.00%
% % Position6@Housing-1
% % % Position6@Housing-1 X Component (i,17)
x17=icdf('Normal',rand(1,N),0 ,0.5);
% % % Position6@Housing-1 Y Component (i,18)
x018=sqrt(abs(0.5^2-x17.^2));
x18=icdf('Normal',rand(1,N),0 ,x018);
% % Diameter6@Housing-1 (i,19)
x19=icdf('Normal',rand(1,N),0 ,0.5);

% Boss2@Worm Gear Shaft-1=8.00%
% % Position1@Worm Gear Shaft-1
% % % Position1@Worm Gear Shaft-1 X Component (i,20)
x20=icdf('Normal',rand(1,N),0 ,0.5);
% % % Position1@Worm Gear Shaft-1 Y Component (i,21)
x021=sqrt(abs(0.5.^2-x20.^2));
x21=icdf('Normal',rand(1,N),0 ,x021);
% % Diameter2@Worm Gear Shaft-1 (i,22)
x22=icdf('Normal',rand(1,N),12.5 ,0.1);

xlswrite(filename,N, sheet, 'A1') % N Sayısını 1x1 e
yazdırıyoruz

xlswrite(filename,x1', sheet, 'B2') % Modelde kullanılan
xlswrite(filename,x2', sheet, 'C2') % Modelde kullanılan
xlswrite(filename,x3', sheet, 'D2') % Modelde kullanılan
xlswrite(filename,x4', sheet, 'E2') % Modelde kullanılan
xlswrite(filename,x5', sheet, 'F2') % Modelde kullanılan
xlswrite(filename,x6', sheet, 'G2') % Modelde kullanılan
xlswrite(filename,x7', sheet, 'H2') % Modelde kullanılan
xlswrite(filename,x8', sheet, 'I2') % Modelde kullanılan
xlswrite(filename,x9', sheet, 'J2')
xlswrite(filename,x10', sheet, 'K2')
xlswrite(filename,x11', sheet, 'L2')
xlswrite(filename,x12', sheet, 'M2')
xlswrite(filename,x13', sheet, 'N2')
xlswrite(filename,x14', sheet, 'O2')
xlswrite(filename,x15', sheet, 'P2')
xlswrite(filename,x16', sheet, 'Q2')
xlswrite(filename,x17', sheet, 'R2') % Modelde kullanılan

```

```
xlswrite(filename,x18',sheet,'S2') % Modelde kullanılan  
xlswrite(filename,x19',sheet,'T2') % Modelde kullanılan  
xlswrite(filename,x20',sheet,'U2') % Modelde kullanılan  
xlswrite(filename,x21',sheet,'V2') % Modelde kullanılan  
xlswrite(filename,x22',sheet,'W2') % Modelde kullanılan  
  
toc;  
end
```



## EK 10 gbox\_macro.swp macro dosyası ve modulleri

<b>gbox_macro.swp</b>	
<b>Main Module</b>	
Option Explicit	
Dim swApp As SldWorks.SldWorks	
Dim swModel As SldWorks.ModelDoc2	
Dim swSelMgr As SldWorks.SelectionMgr	
Public Const Location = "C:\Users\Utku\Documents\thesis\live\"	
' Başka bir klasörde çalıştırıldığında	
Public Const DataFile = Location & "data10k.xlsx"	
Public Const WGSpart = Location &	
"solid_tolanalyst\model_changed\worm gear shaft.sldprt"	
Public Const RCPpart = Location &	
"solid_tolanalyst\model_changed\round cover plate.sldprt"	
Public Const Hpart = Location &	
"solid_tolanalyst\model_changed\housing.sldprt"	
Public Const GBassy = Location &	
"solid_tolanalyst\model_changed\gear box.sldasm"	
Public mStatus As String	' Ölçüm durumu hakkında bilgi
Public Const intN = 1000	' Test sayısı Excel ile aynı olmalı
Public dblData(intN, 22) As Double	' Excel verileri için dizi
Public dblGap As Double	' Excel kaydı için ölçülen değer
Public i As Integer	' Döngü için satır sayacı
Public j As Integer	' Döngü için sütun sayacı
Sub main()	
Dim boolstatus As Boolean	
Dim errors As Long	
Dim warnings As Long	
Set swApp = Application.SldWorks	
''' DOSYALARI AÇ '''	
' Worm Gear Shaft PART	
Dim WormGearShaft As String	
Set swModel = swApp.OpenDoc6(WGSpart,	
swDocumentTypes_e.swDocPART, swOpenDocOptions_Silent, "",	
errors, warnings)	
'Debug.Print errors & warnings	
WormGearShaft = swModel.GetTitle	
Debug.Print WormGearShaft + " açtım."	
' Round Cover Plate PART	
Dim RoundCoverPlate As String	

```

Set swModel = swApp.OpenDoc6(RCPpart,
swDocumentTypes_e.swDocPART, swOpenDocOptions_Silent, "",
errors, warnings)
'Debug.Print errors & warnings
RoundCoverPlate = swModel.GetTitle
Debug.Print RoundCoverPlate + " açtım."

' Housing PART
Dim Housing As String
Set swModel = swApp.OpenDoc6(Hpart,
swDocumentTypes_e.swDocPART, swOpenDocOptions_Silent, "",
errors, warnings)
'Debug.Print errors & warnings
Housing = swModel.GetTitle
Debug.Print Housing + " açtım."

' Gear Box ASSY
Dim GearBox As String
Set swModel = swApp.OpenDoc6(GBassy,
swDocumentTypes_e.swDocASSEMBLY, swOpenDocOptions_Silent, "",
errors, warnings)
'Debug.Print errors & warnings
GearBox = swModel.GetTitle
Debug.Print GearBox + " açtım."
    Call MeasureFaces(1)
Debug.Print ("Varsayılan boşluk " & dblGap & " mm")
    Call SaveDataToExcel(0, 0) ' ilk sıfır tricky ikinci sütun
için

' Excel den verileri al
Call GetDataFromExcel

''' DEĞİŞİKLİKLERİ UYGULA '''
Set swSelMgr = swModel.SelectionManager

    For i = 1 To intN ' normalde 5 kısa sürsün diye
yaptık.
        'For j = 1 To 22 ' normalde 8 kısa sürsün diye
yaptık.
            mStatus = ""

            ' Boss2@Worm Gear Shaft-1=8.00%
swApp.ActivateDoc WormGearShaft
            'Debug.Print errors
            Set swModel = swApp.ActiveDoc
            ' Position1@Worm Gear Shaft-1
swModel.Parameter("D2@Sketch2").SystemValue = (dblData(i, 20)
/ 1000)
swModel.Parameter("D3@Sketch2").SystemValue = (dblData(i, 21)
/ 1000)

                swModel.EditRebuild3
swModel.ClearSelection

```

```

        ' Diameter2@Worm Gear Shaft-1
swModel.Parameter("D1@Sketch2").SystemValue = (dblData(i, 22)
/ 1000)
        swModel.EditRebuild3
swModel.ClearSelection
        ' Hole Pattern1@Round Cover Plate-1=19.20%
        ' Hole Pattern1@Round Cover Plate-2=19.20%
swApp.ActivateDoc RoundCoverPlate
        'Debug.Print errors
        Set swModel = swApp.ActiveDoc
        ' Position1@Round Cover Plate
swModel.Parameter("D4@Sketch1").SystemValue = ((90 / 1000) +
(dblData(i, 5) / 1000))
        swModel.EditRebuild3
swModel.ClearSelection
        ' Diameter1@Round Cover Plate
swModel.Parameter("D1@Sketch4").SystemValue = (dblData(i, 7) /
1000)
        swModel.EditRebuild3
swModel.ClearSelection
        ' Diameter3@Round Cover Plate
swModel.Parameter("D3@Sketch1").SystemValue = (dblData(i, 8) /
1000)
        swModel.EditRebuild3
swModel.ClearSelection
        ' Hole Pattern10@Housing-1=27.20%
swApp.ActivateDoc Housing
        'Debug.Print errors
        Set swModel = swApp.ActiveDoc
        ' Position8@Housing-1 (Front)
Because drilled front to end
swModel.Parameter("D1@Sketch9").SystemValue = ((90 / 1000) +
(dblData(i, 1) / 1000))
        swModel.EditRebuild3
swModel.ClearSelection
        ' Diameter3@Housing-1 (Front)
Because drilled front to end
swModel.Parameter("D1@Sketch5").SystemValue = (dblData(i, 3) /
1000)
        swModel.EditRebuild3
swModel.ClearSelection
        ' Diameter8@Housing-1 (Front)
Because drilled front to end
swModel.Parameter("D2@Sketch9").SystemValue = (dblData(i, 4) /
1000)
        swModel.EditRebuild3
swModel.ClearSelection
        ' Simple Hole3@Housing-1=14.40%
        ' Position3@Housing-1
        ' Same as Position8@Housing-1

```

```

' Diameter1@Housing-1
' Same as Diameter3@Housing-1
' Diameter3@Housing-1
' Same as Diameter3@Housing-1 (Back)
Because drilled front to end, Changed simultaneously
' Simple Hole8@Housing-1=12.00%
' Position6@Housing-1
' Same as Position8@Housing-1 (Back)
Because drilled front to end, Changed simultaneously
' Diameter6@Housing-1
' Same as Diameter8@Housing-1 (Back)
Because drilled front to end, Changed simultaneously

' Gear Box Assembly
swApp.ActivateDoc GearBox
' Debug.Print errors
Set swModel = swApp.ActiveDoc
swModel.EditRebuild3
Call MeasureFaces(0)
Call SaveDataToExcel(i, 0)
' Debug.Print ("Merkezler " & dblGap & "
mm")

Call MeasureFaces(1)
Call SaveDataToExcel(i, 1)
' Debug.Print ("En az " & dblGap & " mm")
Call MeasureFaces(2)
Call SaveDataToExcel(i, 2)
' Debug.Print ("En fazla " & dblGap & "
mm")

' Next j
Next i
MsgBox (i - 1 & " test için sonuçlar alındı.")
End Sub

```

#### **Get\_Excel**

```

' *****
' Preconditions:
'
' Postconditions: None.
' *****
Sub GetDataFromExcel()
Dim xlApp As Excel.Application
Dim xlWB As Excel.Workbook
Dim xlWS As Excel.Worksheet

Set xlApp = CreateObject("Excel.Application")
Set xlWB = xlApp.Workbooks.Open(DataFile, ReadOnly:=True)
Set xlWS = xlWB.Worksheets("Sayfa1")
xlApp.Visible = False
xlApp.DisplayAlerts = False

'verileri diziye aktarma
Dim firstRow As Integer

```

```

Dim firstCol As Integer
firstRow = 2
firstCol = 2

For i = 1 To intN
    For j = 1 To 22
        dblData(i, j) = xlWS.Cells(firstRow + i -
1, firstCol + j - 1).Value
        ' Debug.Print i & "," & j & " " &
dblData(i, j)
    Next j
Next i

'xlApp.DisplayAlerts = True
'xlApp.Visible = True
' Kapat Gitsin
xlWB.Close
xlApp.Quit
End Sub

```

#### Measure Faces

```

'*****
' Preconditions: Model is open.
'
' Postconditions: None.
'*****
Sub MeasureFaces(intSet As Integer)

Dim swApp As SldWorks.SldWorks
Dim Part As ModelDoc2
Dim Measure As Measure
Dim boolstatus As Boolean
Dim boolstatus2 As Boolean
Set swApp = Application.SldWorks
Set Part = swApp.ActiveDoc

' Modeli izometrik olarak ekrana sığdır
Part.ShowNamedView2 "*İzometrik", 7
Part.ViewZoomtofit2

' İki yüzey seç
boolstatus = Part.Extension.SelectByID2("", "FACE",
3.1511233566448E-03, 7.23974921572221E-02, 6.02790212595892E-
02, False, 0, Nothing, 0)
boolstatus2 = Part.Extension.SelectByID2("", "FACE", -
7.8403926350461E-03, 6.88250324188857E-02, 4.88876870994659E-
02, True, 0, Nothing, 0)
'boolstatus = Part.Extension.SelectByID2("Face2@Cut-
Extrudel@Round Cover Plate<2>", "FACE", 0, 0, 0, False, 0,
Nothing, 0)
'boolstatus2 = Part.Extension.SelectByID2("Facel@Boss-
Extrudel@Worm Gear Shaft<2>", "FACE", 0, 0, 0, True, 0,
Nothing, 0)

```

```

    If (boolstatus = False Or boolstatus2 = False) Then
        mStatus = "Can not selected entities."
    End If
' Ölç
Set Measure = Part.Extension.CreateMeasure
Measure.ArcOption = intSet 'Can set this to 0, 1, or 2
                        ' 0 - Center to Center
                        ' 1 - Minimum Distance
                        ' 2 - Maximum Distance

Dim strDisplay As String

boolstatus = Measure.Calculate(Nothing)
If (boolstatus) Then
    If (Not (Measure.Length = -1)) Then
Debug.Print "Length: " & Measure.Length
    End If

    If (Not (Measure.Area = -1)) Then
Debug.Print "Area: " & Measure.Area
    End If

    If (Not (Measure.ArcLength = -1)) Then
Debug.Print "Arc length: " & Measure.ArcLength
    End If

    If (Not (Measure.ChordLength = -1)) Then
Debug.Print "Chord length: " & Measure.ChordLength
    End If

    If (Not (Measure.Diameter = -1)) Then
Debug.Print "Diameter: " & Measure.Diameter
    End If

    If (Not (Measure.Radius = -1)) Then
Debug.Print "Radius: " & Measure.Radius
    End If

    If (Not (Measure.Perimeter = -1)) Then
Debug.Print "Perimeter: " & Measure.Perimeter
    End If

    If (Not (Measure.x = -1)) Then
Debug.Print "X coordinate: " & Measure.x
    End If

    If (Not (Measure.Y = -1)) Then
Debug.Print "Y coordinate: " & Measure.Y
    End If

    If (Not (Measure.Z = -1)) Then
Debug.Print "Z coordinate: " & Measure.Z

```

```

End If

If (Not (Measure.DeltaX = -1)) Then
'   Debug.Print "DeltaX: " & Measure.DeltaX
End If

If (Not (Measure.DeltaY = -1)) Then
'   Debug.Print "DeltaY: " & Measure.DeltaY
End If

If (Not (Measure.DeltaZ = -1)) Then
'   Debug.Print "DeltaZ: " & Measure.DeltaZ
End If

If (Not (Measure.Angle = -1)) Then
Debug.Print "Angle: " & Measure.Angle
End If

If (Not (Measure.CenterDistance = -1)) Then
'   Debug.Print "Center distance: " &
Measure.CenterDistance
End If

If (Not (Measure.NormalDistance = -1)) Then
Debug.Print "Normal distance: " & Measure.NormalDistance
End If

If (Not (Measure.Distance = -1)) Then
'   Debug.Print "Distance: " & Measure.Distance

    dblGap = Measure.Distance * 1000

End If

If (Not (Measure.TotalLength = -1)) Then
Debug.Print "Total length: " & Measure.TotalLength
End If

If (Not (Measure.TotalArea = -1)) Then
'   Debug.Print "Total Area : " & Measure.TotalArea
End If
If (Measure.IsParallel) Then
Debug.Print "IsParallel : " & Measure.IsParallel
End If

If (Measure.IsIntersect) Then
Debug.Print "Is Intersect : " & Measure.IsIntersect
mStatus = "Is Intersect : " & Measure.IsIntersect
End If

If (Measure.IsPerpendicular = -1) Then
Debug.Print "Is Perpendicular : " & Measure.IsPerpendicular

```

```

End If

If (Not (Measure.Projection = -1)) Then
Debug.Print "Projection : " & Measure.Projection
End If

If (Not (Measure.Normal = -1)) Then
Debug.Print "Normal : " & Measure.Normal
End If

If (Not (Measure.SphericalCenterDistance = -1)) Then
Debug.Print "Spherical Center Distance : " &
Measure.SphericalCenterDistance
End If

If (Measure.IsConcentricSpheres) Then
Debug.Print "Is Concentric Spheres : " &
Measure.IsConcentricSpheres
End If
Else
Debug.Print "Invalid combination of selected entities."
mStatus = "Invalid combination of selected entities."
End If

' Clear Selection
boolstatus = Part.Extension.SelectByID2("", "FACE",
3.1511233566448E-03, 7.23974921572221E-02, 6.02790212595892E-
02, True, 0, Nothing, 0)
boolstatus2 = Part.Extension.SelectByID2("", "FACE", -
7.8403926350461E-03, 6.88250324188857E-02, 4.88876870994659E-
02, True, 0, Nothing, 0)
'boolstatus = Part.Extension.SelectByID2("Face2@Cut-
Extrudel@Round Cover Plate<2>", "FACE", 0, 0, 0, True, 0,
Nothing, 0)
'boolstatus2 = Part.Extension.SelectByID2("Face1@Boss-
Extrudel@Worm Gear Shaft<2>", "FACE", 0, 0, 0, True, 0,
Nothing, 0)
End Sub

```

**Save Excel**

```

!*****
' Preconditions:
'
' Postconditions: None.
!*****
Sub SaveDataToExcel(i, k As Integer)
Dim xlApp As Excel.Application
Dim xlWB As Excel.Workbook
Dim xlWS As Excel.Worksheet

Set xlApp = CreateObject("Excel.Application")
xlApp.Visible = False

```



```

xlApp.DisplayAlerts = False
Set xlWB = xlApp.Workbooks.Open(DataFile)
Set xlWS = xlWB.Worksheets("Sayfa1")

' verileri dosyaya aktarma

Dim firstRow As Integer
Dim firstCol As Integer
firstRow = 2

'For i = 1 To intN      ' Satır değerini girdi
olarak zaten alıyoruz.
'For j = 1 To 22      ' Hep aynı sütuna
kaydedilecek çünkü
' Zaten döngü içinde çağrılıyor
xlWS.Cells(firstRow + i - 1, 24 + k).Value = dblGap
xlWS.Cells(firstRow + i - 1, 28).Value = mStatus
'Next j
'Next i

'xlApp.DisplayAlerts = True
'xlApp.Visible = True
' Kapat Gitsin
xlWB.SaveAs DataFile, AccessMode:=xlExclusive,
ConflictResolution:=Excel.XlSaveConflictResolution.xlLocalSessionChanges
xlWB.Close (True)
xlApp.Quit
End Sub

```

## EK 11 gbox\_rsm fonksiyon dosyası

```

                                gbox_rsm.m
function gbox_rsm( ~ )

clear; clc;           % Clean cache
tic;                 % Time Counter Start
filename='C:\Users\Utku\Documents\thesis\3_thesis\ekler\gbox.x
lsx';
sheet='modelM';
%model='linear'; % Constant and linear terms (the default)
model='interaction'; % Constant, linear, and interaction terms
%model='quadratic'; % Constant, linear, interaction, and
squared terms
%model='purequadratic'; % Constant, linear, and squared terms
Nmodel=20;
Npredict=96;
% Row Col works for model
firstRow = 2;        % Fixed
firstCol = 2;        % Fixed
lastCol=28;         % Covers y
lastRow=Npredict+1; % auto last
firstColletter = xlcolumnletter(firstCol);
lastColletter = xlcolumnletter(lastCol);
modelRange=[firstColletter,num2str(firstRow),':',lastColletter
,num2str(lastRow)];
% Regression Model
% [s,y] = MCS(Nmodel); % Random
% Fixed
% load('sy100k.mat')
        t=xlsread(filename,sheet,modelRange); % import all
excel
        s1=t(1:Nmodel,1:8);           % What you need
        s2=t(1:Nmodel,17:22);        % What you need
s=[s1,s2];           % merge
        y1=t(1:Nmodel,24:24);        % Minimum Measured Col A
        y2=t(1:Nmodel,27:27);        % Opposite Measured Col B
y=[y1;y2];          % merge
y = datasample(y,Nmodel,'Replace',false);
        stats = regstats(y, s, model);
        b = stats.beta; % Model coefficients
% Prediction
% [s,~]=MCS(Npredict); % Random
% Fixed
% load('sy100k.mat')
        ss1=t(1:Npredict,1:8);       % What you need
        ss2=t(1:Npredict,17:22);    % What you need
        ss=[ss1,ss2];               % merge
        SS = x2fx(ss, model);
        YY = SS*b; % Predictions of S
        [mean_y, std_y, skw_y, krt_y, max, min, ~] = statistic(YY,
Npredict);

```

```

time=toc;
% cellRange =
[firstCol,num2str(firstRow),':',lastCol,num2str(lastRow)]
sheet='gbox_rsm';
firstColletter = xlcolumnletter(firstCol);
cellRange = [firstColletter,num2str(firstRow)];
xlswrite(filename,Nmodel,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,{model},sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,mean_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,std_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,skw_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,krt_y,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,max,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,min,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,' ',sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,time,sheet,cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,Npredict,sheet,cellRange)

winopen(filename);

toc      % Time Counter End
end

```

## EK 12 gbox\_rand.m fonksiyon dosyası

```

gbox.km.m
function gbox_km( ~ )

clear; clc;          % Clean cache
tic;                % Time Counter Start
filename='C:\Users\Utku\Documents\thesis\3_thesis\ekler\gbox.x
lsx';
sheet='modelM';
regpol=2;
Nmodel=20;
Npredict=96;
% Row Col works for model
firstRow = 2;      % Fixed
firstCol = 2;      % Fixed
lastCol=28;        % Covers y
lastRow=Npredict+1; % auto last
firstColletter = xlcolumnletter(firstCol);
lastColletter = xlcolumnletter(lastCol);
modelRange=[firstColletter,num2str(firstRow),':',lastColletter
,num2str(lastRow)];
% Regression Model
% [s,y] = MCS(Nmodel); % Random
% Fixed
% load('sy100k.mat')
t=xlsread(filename,sheet,modelRange); % import all
excel
s1=t(1:Nmodel,1:8); % What you need
s2=t(1:Nmodel,17:22); % What you need
s=[s1,s2]; % merge
y1=t(1:Nmodel,24:24); % Minimum Measured Col A
y2=t(1:Nmodel,27:27); % Opposite Measured Col B
y=[y1;y2];
y = datasample(y,Nmodel,'Replace',false);
[dmodel,~]=dacefit(s, y, @regpoly1, @corrGauss, 2);
% Prediction
% [s,~]=MCS(Npredict); % Random
% Fixed
% load('sy100k.mat')
ss1=t(1:Npredict,1:8); % What you need
ss2=t(1:Npredict,17:22); % What you need
ss=[ss1,ss2]; % merge
YY=predictor(ss, dmodel);
[mean_y, std_y, skw_y, krt_y, max, min, ~] = statistic(YY,
Npredict);

time=toc;
% cellRange =
[firstCol,num2str(firstRow),':',lastCol,num2str(lastRow)]
sheet='gbox_km';
firstColletter = xlcolumnletter(firstCol);

```

```

cellRange = [firstColletter,num2str(firstRow)];
xlswrite(filename,Nmodel, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,{regpol}, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,mean_y, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,std_y, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,skw_y, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,krt_y, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,max, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,min, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,' ', sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,time, sheet, cellRange)
firstRow=firstRow+1; cellRange =
[firstColletter,num2str(firstRow)];
xlswrite(filename,Npredict, sheet, cellRange)

winopen(filename);

toc      % Time Counter End
end

```

## KAYNAKLAR

- [1] Drake, Paul J. Dimensioning and tolerancing handbook. New York: McGraw-Hill, 1999.
- [2] "tolerans" erişim adresi: [http://www.tdk.gov.tr/index.php?option=com\\_gts&arama=gts](http://www.tdk.gov.tr/index.php?option=com_gts&arama=gts), erişim tarihi: 20 Kasım 2014
- [3] Srinivasan, R. S. "A theoretical framework for functional form tolerances in design for manufacturing." (1994).
- [4] ASME Y14.5M-1994 (ed.), 1995, Dimensioning and Tolerancing: ASME Y14.5M-1994, ASME Standard Committee Y14: Engineering Drawing and Related Documentation Practices (ASME)
- [5] ISO, PNEN. "1101: 2012." Geometrical Product Specifications (GPS)–Geometrical tolerancing–Tolerances of form, orientation, location and run-out.
- [6] HENZOLD, Georg. Geometrical dimensioning and tolerancing for design, manufacturing and inspection: a handbook for geometrical product specification using ISO and ASME standards. Butterworth-Heinemann, 2006.
- [7] Hong, Y. S., and T. C. Chang. "A comprehensive review of tolerancing research." International Journal of Production Research 40.11 (2002): 2425-2459.
- [8] Chase, Kenneth W., and Alan R. Parkinson. "A survey of research in the application of tolerance analysis to the design of mechanical assemblies." Research in Engineering design 3.1 (1991): 23-37.
- [9] Prisco, Umberto, and Giuseppe Giorleo. "Overview of current CAT systems." Integrated Computer-Aided Engineering 9.4 (2002): 373-387.
- [10] Fortini, Earlwood T., and E. T. Fortini. Dimensioning for interchangeable manufacture. New York: Industrial Press, 1967.
- [11] Greenwood, W. H., and K. W. Chase. "A new tolerance analysis method for designers and manufacturers." Journal of Manufacturing Science and Engineering 109.2 (1987): 112-116.
- [12] Mansoor, E. M. "The application of probability to tolerances used in engineering designs." Proceedings of the Institution of Mechanical Engineers 178.1 (1963): 29-39.
- [13] Zhang, C., and H-P. BEN WANG. "Tolerance analysis and synthesis for cam mechanisms." THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH 31.5 (1993): 1229-1245.

- [14] Chase, Kenneth W., Jinsong Gao, and Spencer P. Magleby. "General 2-D tolerance analysis of mechanical assemblies with small kinematic adjustments." *Journal of Design and Manufacturing* 5 (1995): 263-274.
- [15] Gao, Jinsong, Kenneth W. Chase, and Spencer P. Magleby. "Comparison of assembly tolerance analysis by the direct linearization and modified Monte Carlo simulation methods." (1995).
- [16] Greenwood, W. H., and K. W. Chase. "Worst case tolerance analysis with nonlinear problems." *Journal of Manufacturing Science and Engineering* 110.3 (1988): 232-235.
- [17] Lin, Shui-Shun, Hsu-Pin Wang, and Chun Zhang. "Statistical tolerance analysis based on beta distributions." *Journal of Manufacturing Systems* 16.2 (1997): 150-158.
- [18] Nigam, Swami D., and Joshua U. Turner. "Review of statistical approaches to tolerance analysis." *Computer-Aided Design* 27.1 (1995): 6-15.
- [19] Singh, P. K., P. K. Jain, and S. C. Jain. "Important issues in tolerance design of mechanical assemblies. Part 1: tolerance analysis." *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 223.10 (2009): 1225-1247.
- [20] Drake, Paul. "Traditional approaches to analyzing mechanical tolerance stacks." *Dimensioning and Tolerancing Handbook*, New York: McGraw-Hill 9 (1999): 1-39.
- [21] Yang, Kai, and Basem S. El-Haik. *Design for six sigma*. New York: McGraw-Hill, 2003.
- [22] "Monte Carlo Method." Wikipedia. Wikimedia Foundation, 24 Nov. 2014. Web. 24 Nov. 2014. <[http://en.wikipedia.org/wiki/Monte\\_Carlo\\_method](http://en.wikipedia.org/wiki/Monte_Carlo_method)>.
- [23] Casella, George, and Christian P. Robert. "Monte Carlo statistical methods." (1999).
- [24] MATLAB. Vers. R2014a. US: MathWorks, 2014. Program documentation.
- [25] Barton, Russell R. "Metamodels for simulation input-output relations." *Proceedings of the 24th conference on Winter simulation*. ACM, 1992.
- [26] Han, Zhong-Hua, and Ke-Shi Zhang. "Surrogate-Based Optimization." *Intec Book, Real-World Application of Genetic Algorithm*. InTech (2012): 343-362.
- [27] Myers, Raymond H., Douglas C. Montgomery, and Christine M. Anderson-Cook. "Response surface methodology: process and product optimization using designed experiments." (2009).

- [28] Bozan, Aslı, and Ayhan Kural. Optimization of Laser Welding Seam Geometry Using Response Surface Methodology (RSM). Thesis. İTÜ Fen Bilimleri Enstitüsü, 2012.
- [29] Bradley, Nuran. The response surface methodology. Thesis. Diss. Indiana University South Bend, 2007.
- [30] Kaymaz, Irfan. "Application of kriging method to structural reliability problems." *Structural Safety* 27.2 (2005): 133-151.
- [31] Simpson, Timothy W., et al. Comparison of response surface and kriging models in the multidisciplinary design of an aerospike nozzle. Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, 1998.
- [32] Lophaven SN, Nielsen HB, Søndergaard J. DACE, a Matlab Kriging Toolbox v.2.0, 2002.
- [33] Marc Gentile, Guldariya, Nurbaeva, Frederic Courbin, Georges Meylan "Kriging Interpolation Application to the GREAT10 Star Challenge." Web. 20 Nov. 2014.  
<[https://cmml2011.wikispaces.com/file/view/Kriging\\_G10\\_Star\\_Challenge\\_1.1.pdf](https://cmml2011.wikispaces.com/file/view/Kriging_G10_Star_Challenge_1.1.pdf)>
- [34] Wu, Yanyan, Jami J. Shah, and Joseph K. Davidson. "Computer modeling of geometric variations in mechanical parts and assemblies." *Journal of Computing and Information Science in Engineering* 3.1 (2003): 54-63.
- [35] TolAnalyst. Program documentation. SOLIDWORKS Web Help. Vers. 2015 SP01. Dassault Systèmes SolidWorks Corporation, Web. 12 Sept. 2015.  
<[http://help.solidworks.com/2015/English/SolidWorks/SWHelp\\_List.html?id=3a75c22cbbd241f49657a1fa04128d23#Pg0](http://help.solidworks.com/2015/English/SolidWorks/SWHelp_List.html?id=3a75c22cbbd241f49657a1fa04128d23#Pg0)>.



## ÖZGEÇMİŞ

### Kişisel Bilgiler

**Soyadı, adı** : BAYRAM, Utku  
**Uyruğu** : T.C.  
**Doğum tarihi ve yeri** : 11.07.1987 Ankara  
**Medeni hali** : Bekar  
**Telefon** : 0533 413 4365  
**E-posta** : utkubym@gmail.com

### Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek Lisans	Makina Mühendisliği TOBB Ekonomi ve Teknoloji Üniversitesi	2014
Lisans	Makina Mühendisliği TOBB Ekonomi ve Teknoloji Üniversitesi	2011

### İş Deneyimi

Yıl	Yer	Görev
2011-2013	TCI -Türk Kabin İçi Sistemleri	Yapısal Tasarım Mühendisi

### Yabancı Dil

İngilizce (B2), İspanyolca (A1)

### Yayınlar

Bayram U., Acar E., Tolerance Analysis with Multiple Surrogate Models. *International Conference on Computational and Experimental Science and Engineering (ICCESEN - 2014)*. 25-29 October 2014, Antalya, Turkey