

**CNC TAKIM TEZGAHLARI İÇİN KULLANILAN PARÇA
PROGRAMLARININ DOĞRULANMASI VE OPTİMİZASYONU**

Sevilay TURGUT

**YÜKSEK LİSANS TEZİ
MAKİNE EĞİTİMİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

HAZİRAN 2010

ANKARA

Sevilay TURGUT tarafından hazırlanan CNC TAKIM TEZGAHLARI İÇİN KULLANILAN PARÇA PROGRAMLARININ DOĞRULANMASI VE OPTİMİZASYONU adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Yunus KAYIR

Tez Danışmanı, Makine Eğitimi Anabilim Dalı

Tarih : 24 / 05 / 2010

Bu çalışma, jürimiz tarafından oy birliği ile Makine Eğitimi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Doç. Dr. İhsan Korkut

Makine Eğitimi, Gazi Üniversitesi

Doç. Dr. Hüdayim Başak

Makine Eğitimi, Gazi Üniversitesi

Yrd. Doç. Dr. Hakan Dilipak

Makine Eğitimi, Gazi Üniversitesi

Yrd. Doç. Dr. Neşet Akar

Metal Eğitimi , Gazi Üniversitesi

Yrd. Doç. Dr. Yunus Kayır

Makine Eğitimi, Gazi Üniversitesi

Tarih: 24 / 05 / 2010

Bu tez ile G.Ü. Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onamıştır.

Prof. Dr. Bilal TOKLU

Fen Bilimleri Enstitüsü Müdürü

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.

Sevilay TURGUT

**CNC TAKIM TEZGAHLARI İÇİN KULLANILAN PARÇA
PROGRAMLARININ DOĞRULANMASI VE OPTİMİZASYONU
(Yüksek Lisans Tezi)**

Sevilay TURGUT

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Mayıs 2010

ÖZET

Bu çalışmada, CNC takım tezgahları için CAM programları ile oluşturulan parça programları kullanılarak, AutoCAD ortamında takım yolu çizdirilmesine ve değerlendirilmesine yönelik bir sistem geliştirilmiştir. AutoCAD grafik ekranında takım yolları çizilirken; kullanılan takımlar, talaş derinliği, talaş alma hareketleri, boş hareketler, vb. özellikler dikkate alınmıştır. Geliştirilen sistem ile CNC parça programlarının doğruluğu, AutoCAD programının grafik özelliklerinin kullanılarak kontrol edilebilmektedir. AutoCAD ortamında oluşturulan takım yolları incelenerek; parçanın kaba işleme operasyonlarında işleme hassasiyetini etkilemeyen çok küçük takım yollarının belirlenmesine gidilmiştir. Belirlenen takım yollarının, kullanıcı etkileşimli olarak çıkarılabilmesi sağlanarak takım yolu optimizasyonu yapılmıştır. Böylelikle tezgah titreşimlerinin azalmasıyla birlikte, takım ömründe iyileşme ve parça işleme zamanında azalma sağlanmıştır. Sistemin geliştirilmesinde, AutoCAD grafik ara yüzünü kullanan Visual Basic Application ve Activex uygulamalar kullanılmıştır.

Bilim Kodu : 708. 3.028
Anahtar Kelimeler : Takım yolu doğrulama, AutoCAD VBA
Sayfa Adedi : 108
Tez Yöneticisi : Yrd. Doç. Dr. Yunus KAYIR

**VERIFICATION AND OPTIMIZATION OF PART PROGRAMS USED CNC
MILLING MACHINE**

(M.Sc. Thesis)

Sevilay TURGUT

**GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY**

June 2010

ABSTRACT

In this study, a system has been developed for modeling and evaluating cutting tools paths in AutoCAD program by using CNC part programs generated with CAM programs for CNC milling machines. Tools paths verification and optimization could be done for the modeling tools paths.

Vories characteristics, such as Cutting depth, feedrate and rapid paths, etc. were taken into consideration for modeling tools paths in AutoCAD program. Verification of CNC part programs by using capabilities of AutoCAD graphic properties. The modeled tools paths in AutoCAD program were investigated for too small tools paths that are not important for accuracy in milling part surface roughly. A method were used for removing the too small tool paths with user interface.

The removing of the small tool paths are provided reduction of the CNC milling vibration and cutting time decrease and tool life increase. System development, AutoCAD graphic interface that uses the Visual Basic Application and Activex applications are used.

Science Code : 708. 3.028

Key Word : Tool path verification, AutoCAD VBA

Page Number : 108

Adviser : Yrd. Doç. Dr. Yunus KAYIR

TEŞEKKÜR

Çalışmalarım boyunca değerli görüş ve katkılarıyla beni yönlendiren danışmanım Sayın Yrd. Doç. Dr. YUNUS KAYIR'a, bu tez çalışmam sırasında her türlü yardım ve desteğini aldığım sevgili eşim Mustafa TURGUT'a, değerli arkadaşlarım Hümeysra KAYA ve Hüseyin İNAN'a ve bu günlere gelmemde maddi ve manevi destekleriyle beni yalnız bırakmayan çok değerli aileme sonsuz teşekkür ederim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ.....	xii
SİMGELER VE KISALTMALAR	xvii
1. GİRİŞ	1
2. LİTERATÜR ARAŞTIRMASI	3
3. CNC TAKIM TEZGAHLARI.....	9
3.1. Kontrol Üniteleri.....	11
3.2. Sayısal Denetim Veri Yapısı	12
3.3. Hareket Tipleri.....	15
3.3.1. Doğrusal hareket	15
3.3.2. Dairesel hareket.....	17
3.4. Çevrimler	18
4. CAD/CAM.....	19
4.1. Bilgisayar Destekli Tasarım (CAD)	19
4.2. Bilgisayar Destekli İmalat (CAM)	21
4.2.1. Bilgisayar destekli imalat basamakları	23
5. VISUAL BASIC VE VISUAL BASIC APPLICATION	26
5.1. Visual Basic (VB).....	26
5.2. Visual Basic Application (VBA).....	27
5.2.1. VBA yöneticisi ile projeleri düzenleme.....	28
5.3. VBA’da Örnek Çalışma.....	34
5.4. Kullanıcı Formlarının VBA ‘da Tasarlanması	36

	Sayfa
5.4.1. Çalışma sayfaları oluşturma.....	36
5.4.2. Kontrol ekleme ve ayarlama	37
5.4.3. Özelliklerin ayarlanması	37
5.4.4. Kontrol çeşitleri.....	38
6. CNC PARÇA PROGRAMLARI İÇİN TAKIM YOLU DOĞRULAMASI VE OPTİMİZASYONU	41
6.1. CNC parça program bilgilerinin çıkarılması	43
6.2. CNC Parça Programının Doğrulanması	44
6.2.1. Sayısal denetim takım yolunun AutoCAD yazılımında çizdirilmesi.....	45
6.3. CNC Parça Programının Optimizasyonu.....	45
7. CNC PARÇA PROGRAMLARI İÇİN TAKIM YOLU DOĞRULAMASI VE OPTİMİZASYON YÖNTEMİNDE ÖRNEK UYGULAMALAR	53
7.1. Örnek Uygulama 1.....	53
7.2 Örnek Uygulama 2.....	67
7.3. Örnek Uygulama 3.....	71
8. BİLGİSAYAR DESTEKLİ TAKIM YOLU DOĞRULAMASI VE OPTİMİZASYONU PROGRAMININ KULLANILMASI.....	74
8.1. Programın Yüklenmesi ve Kullanılması.....	74
8.2. Sayısal Denetim Takım Yolu Verisinin Seçilmesi ve Gösterilmesi.....	76
8.3. Bitmiş Parçanın Açılması	77
8.4. Sayısal Verinin Dönüştürülmesi.....	78
8.5. Takım Yolunun Çizdirilmesi	79
8.5.1. Takımlara göre çizim yapılması.....	79
8.5.2. Takım yolu boylarına göre çizim yapılması	81
8.5.3. Hareket tipine göre çizim yapılması	83
8.6. Hesaplamaların Gösterilmesi.....	85
8.7. Takım Yolu Optimizasyonu İşlemi	86
9. SONUÇ VE ÖNERİLER.....	88
KAYNAKLAR	89
EKLER.....	91

Sayfa

EK-1 Geliştirilen programın komut satırları.....	92
ÖZGEÇMİŞ	108

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Örnek bir CNC program.....	13
Çizelge 3.2. Sayısal denetim kontrol ünitesi kontrol parametreleri	14
Çizelge 3.3. Bazı G kodları (Fanuc kontrol sistemine göre) ve açıklamaları.....	15
Çizelge 3.4. Bazı M yardımcı fonksiyonları ve açıklamaları	15
Çizelge 5.1. öğrencinin aldığı not aralığına uygun bir mesaj veren program örneği .	27
Çizelge 6.1. Takım yolunun boylarına göre gruplandırılması.....	48
Çizelge 6.2. Takım yolu boyu	49
Çizelge 6.3. Farklı işleme toleranslarında, takım yollarının takım yolu boyuna göre gruplandırılması	51
Çizelge 6.4. Farklı işleme toleranslarında, işleme toleransından küçük takım yolları	52
Çizelge 7.1. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması.....	56
Çizelge 7. 2. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması.....	60
Çizelge 7.3. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması.....	63
Çizelge 7.4. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması.....	66
Çizelge 7.5. Farklı işleme toleranslarındaki varolan CNC kod sonuçlarının karşılaştırılması.....	66
Çizelge 7.6. Farklı işleme toleranslarındaki takım yolları optimizasyon sonuçlarının karşılaştırılması.....	67
Çizelge 7. 7. Farklı işleme toleranslarında takım yollarının, takım yolu boyuna göre gruplandırılması	69
Çizelge 7.8. Farklı işleme toleranslarında takım yollarının, işleme toleranslarından küçük takım yolları	69
Çizelge 7.9. Farklı işleme toleranslarındaki varolan CNC kod sonuçlarının karşılaştırılması.....	70

Çizelge	Sayfa
Çizelge 7.10. Farklı işleme toleranslarındaki takım yolları optimizasyon sonuçlarının karşılaştırılması.....	70
Çizelge 7.11. Farklı işleme toleranslarında takım yollarının, takım yolu boyuna göre gruplandırılması.....	72
Çizelge 7.12. Farklı işleme toleranslarında takım yollarının, işleme toleranslarından küçük takım yolları	72
Çizelge 7.13. Farklı işleme toleranslarındaki varolan CNC kod sonuçlarının karşılaştırılması.....	73

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Wang ve Stori'nin yaptıkları çalışma a) Klasik takım yolu, b) Optimize edilmiş spiral takım yolu	4
Şekil 2.2. İlerleme optimizasyonu stratejilerinin karşılaştırılması	5
Şekil 2.3. Pateloup, Duc ve Ray'ın yaptıkları çalışma, a) Klasik takım yolu, b) Dönüşler BSpline ile yumuşatılmış takım yolu.....	6
Şekil 2.4. SolidCAM yazılımında yüksek hızlı işlemede köşe yumuşatmaları.....	6
Şekil 2.5. Martin ve Cristian'ın yaptıkları çalışma sonucu elde edilen yumuşak spiral takım yolu örnekleri.....	7
Şekil 2.6. Oysu ve Bingül'ün yaptıkları çalışma sonucu takım boşta gezmelerinin minimuma indirilmiş takım yolu	8
Şekil 3.1. CNC takım tezgahı	11
Şekil 3.2. Sayısal denetim kontrol ünitesi	12
Şekil 3.3. Doğrusal hareketler	16
Şekil 3.4. G2 ve G3 dairesel interpolasyonu	17
Şekil 3.5. Örnek çevrim.....	18
Şekil 4.1. CATIA ile yapılan bir tasarım.....	19
Şekil 4.2. Tel kafes model	20
Şekil 4.3. Yüzey model örneği	20
Şekil 4.4. Katı model örneği.....	21
Şekil 4.5. CATIA V5 yazılımında bilgisayar destekli imalat.....	22
Şekil 4.6. Machining Strategy yazılımında bilgisayar destekli imalat	23
Şekil 4.7. Kaba işleme operasyonu.....	24
Şekil 4.8. Ara işleme operasyonu	24
Şekil 4.9. Bitirme işleme operasyonu.....	25
Şekil 5.1. Visual Basic 5 yazılımının genel görünümü	26

Şekil	Sayfa
Şekil 5.2 Komut satırına “VBAMAN” yazılımı.....	28
Şekil 5.3. VBA Manager Düşey Menüsü	29
Şekil 5.4. VBA Manager Penceresi	29
Şekil 5.5. AutoCAD Komut Satırı.....	31
Şekil 5.6. VBA Editörüne Giriş düşey menüsü	31
Şekil 5.7. VBA Editörü	32
Şekil 5.8. Macro Yönetimine Giriş düşey menüsü	33
Şekil 5.9. Yeni bir macro oluşturma.....	34
Şekil 5.10. Add Procedure penceresi.....	35
Şekil 5.11. Kod penceresi	35
Şekil 5.12. Uygulama sonucu AutoCAD çalışma sayfası	36
Şekil 5.13. VBA’da yeni çalışma sayfası	37
Şekil 5.14. VBA’da kullanıcı Formu.....	38
Şekil 5.15. VBA kontrol çeşitleri	40
Şekil 6.1. CNC Parça programları için takım yolu doğrulanması ve optimizasyonu	42
Şekil 6.2. Varolan CNC kodları.....	43
Şekil 6.3. Tablo haline getirilen sayısal denetim takım yolları	44
Şekil 6.4. İşleme toleransı	46
Şekil 6.5. Takım yolunun çıkarılması ile oluşan hata.....	47
Şekil 6.6. İşleme toleransı = 0,1 mm, Yanal kayma mesafesi = $0,5 \times \text{takım çapı} = 0,5 \times 10 = 5 \text{ mm}$, Dalma derinliği = 3 mm olarak oluşturulmuş takım yolu ...	48
Şekil 6.7. Takım yolu hataları	50
Şekil 6.8. Farklı işleme toleranslarında takım yolu boyuna göre takım yolunun grafik olarak gösterilmesi	51
Şekil 7.1. Takım yolu doğrulaması ve optimizasyonu yapılan parça modeli.....	53

Şekil	Sayfa
Şekil 7.2. İşleme toleransı 0.3 mm olan takım yolu sonuçları	54
Şekil 7.3. İşleme toleransı 0.3 mm olan takım yolu	54
Şekil 7.4. İşleme toleransı 0.3 mm'den küçük takım yollarının bulunduğu katmanlar ve takım yollarının gösterilmesi	55
Şekil 7.5. İşleme toleransı 0.3 mm'den küçük takım yollarının bulunması	55
Şekil 7.6. İşleme toleransı 0.3 mm'ye göre optimizasyon yapılmış takım yolu sonuçları.....	56
Şekil 7.7. Var olan CNC kodları ve optimizasyon sonucu oluşturulan CNC kodları karşılaştırılması a) Var olan CNC kodlar, b) Optimizasyon sonrası oluşturulan CNC kodları.....	57
Şekil 7.8. İşleme toleransı 0.2 mm olan takım yolu sonuçları	58
Şekil 7.9. İşleme toleransı 0.2 mm olan takım yolu	58
Şekil 7.10. İşleme toleransı 0.2 mm'den küçük takım yollarının gösterilmesi	58
Şekil 7.11. İşleme toleransı 0.2 mm'den küçük takım yollarının bulunması	59
Şekil 7.12. İşleme toleransı 0.2 mm'ye göre optimizasyon yapılmış takım yolu sonuçları.....	59
Şekil 7.13. İşleme toleransı 0.1 mm olan takım yolu sonuçları.....	60
Şekil 7.14. İşleme toleransı 0.1 mm olan takım yolu	61
Şekil 7.15. İşleme toleransı 0.1 mm'den küçük takım yolları.....	61
Şekil 7.16. İşleme toleransı 0.1 mm'den küçük takım yollarının bulunması	62
Şekil 7.17. İşleme toleransı 0.1 mm'ye göre optimizasyon yapılmış takım yolu sonuçları.....	62
Şekil 7.18. İşleme toleransı 0.05 mm olan takım yolu sonuçları.....	63
Şekil 7.19. İşleme toleransı 0.05 mm olan takım yolu	64
Şekil 7.20. İşleme toleransı 0.05 mm'den küçük takım yolları.....	64
Şekil 7.21. İşleme toleransı 0.05 mm'den küçük takım yollarının bulunması	65

Şekil	Sayfa
Şekil 7.22. İşleme toleransı 0.05 mm'ye göre optimizasyon yapılmış takım yolu sonuçları.....	65
Şekil 7.23. Takım yolu doğrulaması ve optimizasyonu yapılan parça modeli.....	67
Şekil 7.24. İşlem toleransı 0.3 mm olan takım yolu	68
Şekil 7.25. Takım yolu doğrulaması ve optimizasyonu yapılan parça modeli.....	71
Şekil 7.26. İşlem toleransı 0.3 mm olan takım yolu	71
Şekil 8.1. “Load / Unload Application” penceresi	74
Şekil 8.2. Microsoft Visual Basic 5.0 editörü ve takım yolu doğrulama ve optimizasyonu programı ara yüzü.....	75
Şekil 8.3. Takım Yolu Doğrulama ve Optimizasyonu program ara yüzü	76
Şekil 8.4. Sayısal denetim verisinin gösterilmesi	77
Şekil 8.5. Bitmiş parçanın AutoCAD çizim alanında gösterilmesi	78
Şekil 8.6. Sayısal denetim verisinin dönüştürülmesi.....	78
Şekil 8.7. Takım yolu hesaplama raporu	79
Şekil 8.8. Takım Yolu Çizim penceresi.....	79
Şekil 8.9. Takımlara göre çizilmiş örnek takım yolu	80
Şekil 8.10. Takım katmanları	80
Şekil 8.11. T4 takımı ile işlemenin gösterilmesi	81
Şekil 8.12. Takım yolu boyuna göre çizim ayarları	82
Şekil 8.13. Takım yolu boyuna göre takım yolunun çizilmesi.....	82
Şekil 8.14. 0,54 – 0,63 arası boyutlardaki takım yollarının gösterilmesi	83
Şekil 8.15. Hareket tipine göre çizim penceresi	84
Şekil 8.16. Hareket tipine göre çizimlerin gösterilmesi	84
Şekil 8.17. G00 takım yolu hareketleri.....	85
Şekil 8.18. Takım yolu hesaplamaları ve istatistikleri.....	86

Şekil	Sayfa
Şekil 8.19. Takım yolu optimizasyon penceresi.....	86
Şekil 8.20. Takım yolu boyuna göre optimizasyon işlemi	87

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklama
2B	İki Boyut
3B	Üç Boyut
BDT	Bilgisayar Destekli Tasarım
BDÜ	Bilgisayar Destekli Üretim
BSD	Bilgisayarlı Sayısal Denetim
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CNC	Computer Numerical Control
NC	Numerical Control
VBA	Visual Basic for Application
VB	Visual Basic

1. GİRİŞ

Otomotiv sektöründen havacılık sektörüne kadar imalatın her alanında CNC tezgahları kullanılmaktadır.

Takım tezgahlarının işlem süreçlerinde birçok zorluklarla karşılaşmaktadır. Ürün kalitesi operatör yeteneğine ve CAM verisine bağlı olarak değişiklik gösterebilmektedir. İmalat verimliliğini artırabilmek için, takım ömrü, yüzey kalitesi ve işlenebilirlik arasında iyi bir değerlendirmenin yapılması, yapılan değerlendirmenin ardından sonuçların optimize edilmesi gerekmektedir. Günümüzde imalat sanayisine yönelik değişik optimizasyon teknikleri kullanılmaktadır. Bu tekniklerden biri de takım yolunun optimize edilmesidir. Takım yolu optimizasyonu sonucunda toplam takım yolu boyunun azalmasıyla işleme zamanının da azaldığı öngörülmüştür.

Kaynak araştırmalarında takım yolu optimizasyonuna yönelik birçok çalışmanın yapıldığı belirlenmiştir. Bu çalışmalarda geliştirilen programlarda, takım yolu boyuna göre bir değerlendirme yapılmadığı tesbit edilmiştir. Bu sebepten dolayı böyle bir çalışmanın CAM programlarına yardımcı olabileceği ve farklı bir bakış açısı oluşturabileceği düşünülmüştür.

Yapılan bu çalışmada, mevcut CNC programına göre AutoCAD ortamında takım yolu oluşturulmuştur. Takım yolu çizdirme işlemi çeşitli durumlara göre yapılmıştır. Bunlardan bazıları aşağıda sıralanmıştır.

Takım yolunun her takıma göre ayrı ayrı çizdirilmesi

Bu sayede kullanıcı, takım yolunu görerek işlemenin nasıl gerçekleştiği hakkında fikir sahibi olabilmektedir. AutoCAD ortamındaki Layer özelliği kullanılarak, gruplanan takım yollarından yalnızca istenilenler görülebilmektedir.

Hareket boyuna göre takım yolunun gruplandırılması

CNC kodları (G ve M) çıkarılırken istenen işleme toleransına göre iki nokta arasındaki mesafe belirlenmiştir. Bu durum sonucu takım yolu değerinde çok küçük hareketler oluşabilmektedir. Hazırlanan çalışmada kullanıcı, kaba işlemedeki yüzey hassasiyetini etkilemeyecek olan bu küçük hareketleri önleyebilmektedir. Böylelikle de işleme zamanı düşürülerek ve daha verimli bir imalatın gerçekleştirileceği düşünülmektedir.

Ayrıca CNC verileri okunarak; toplam takım yolu boyu, en küçük takım yolu boyu, en büyük takım yolu boyu, CNC program satır sayısı gibi sonuçlar hesaplanmakta ve kullanıcıya rapor edilmektedir. Kullanıcının bu sonuçları değerlendirmesiyle; CNC programının, tezgah hareketleri üzerindeki olumsuz etkilerinin (fazladan yapılan tekrarlar, parça işlenmesinde yüzey kalitesini etkilemeyecek kadar küçük olan hareketler, vb.) giderilmesi sağlanmıştır. Böylelikle, gereksiz hareketlerin CNC programından çıkarılması ile CNC program satır sayısı azalmaktadır. Buna bağlı olarak da toplam takım yolu boyu azalırken toplam takım yolu boyunun doğrudan etkilediği işleme zamanının da azalması öngörülmektedir.

Bu çalışmada Visual Basic for Application yazılımı kullanılarak, AutoCAD programında grafik ara yüzü oluşturulmuştur.

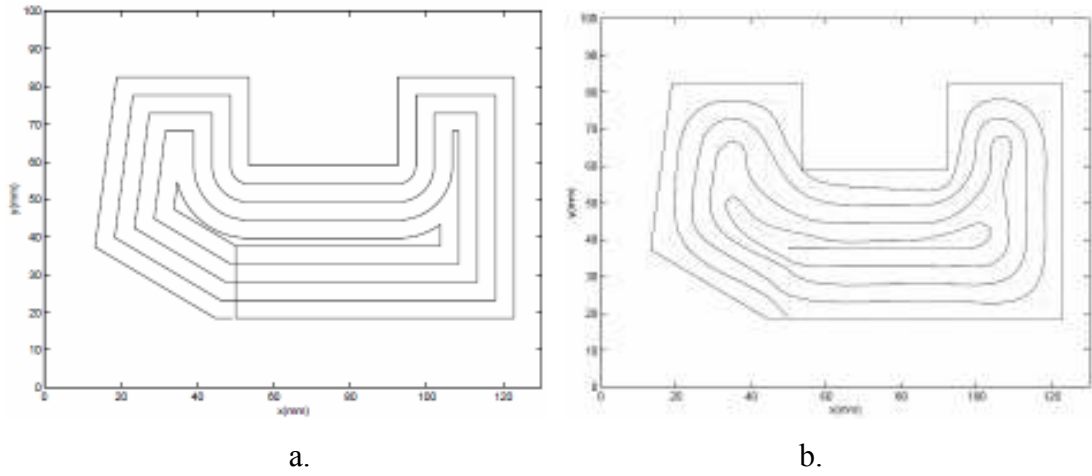
2. LİTERATÜR ARAŞTIRMASI

Sayısal denetim takım yollarının doğrulanması ve optimizasyonu konusunda yapılan çalışmaların bir kısmı aşağıda özetlenmiştir.

Kim, Lee ve Cho yaptıkları çalışmada optimum işleme için kesme kuvvetine bağlı olarak ilerleme optimizasyonu yapmışlardır [1]. Çalışmada takma uçlu parmak freze ele alınmıştır. Kaba işlemlerde ilerleme optimize edilmeden, takma uçlu parmak frezeler yüksek maliyetler getirmektedirler. İlerlemenin optimizasyonuna yönelik yapılan bu çalışma göstermiştir ki, takma uçlu parmak frezelerde işleme zamanı %14 - %44 arasında düşüş göstermiştir. Yapılan çalışmada kesiciye göre belirlenen kesme kuvveti referans alınarak ilerlemeler optimize edilmiştir.

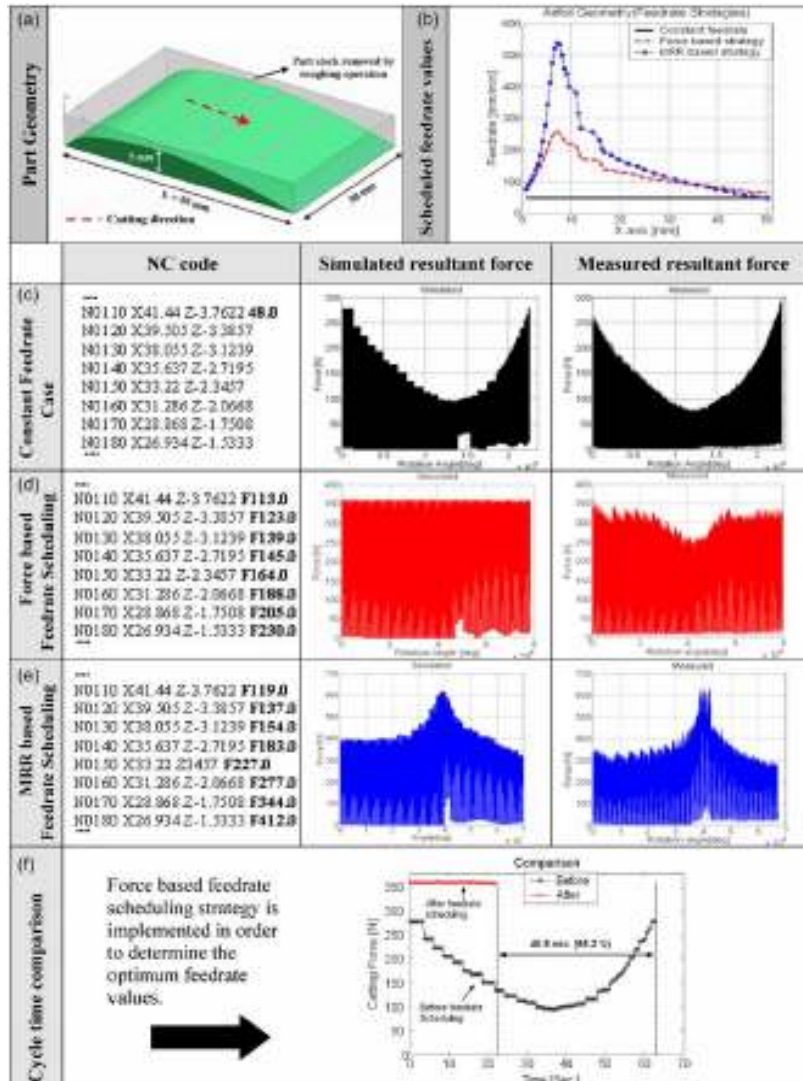
Rabert ve arkadaşları yaptıkları çalışmada her takım hareketi için kesme kuvveti hesaplamışlardır [2]. Kesme kuvveti değişimine paralel olarak ilerlemeyi ve devir sayısını optimize etmişlerdir. Yapılan bu çalışma sonucunda kesme kuvveti sabit tutularak üretim kalitesi arttırılmıştır. İşleme zamanı %13 - %16 arasında azaldığı görülmüştür.

Wang ve Stori yaptıkları çalışmada, 2B'lu takım yollarında, kesiciye gelen kesme kuvvetini referans alarak, takım yolu optimizasyonunu yapmışlardır [3]. Klasik takım yollarında (Şekil 2.1.a), takım yolları doğrusaldır. Bunun sonucu olarak iki takım yolunun birleşimindeki geçiş keskin olacaktır. Bu keskin dönüşte kesicinin işlenen parça ile temas ettiği yüzey artmaktadır. Dolayısıyla, temas yüzeyinin artmasına paralel olarak kesiciye gelen kesme kuvveti de artmaktadır. İşleme esnasında kesici üzerinde oluşan kesme kuvveti farklılıkları; kesici ömrünü, işlenen yüzeyin yüzey kalitesini ve ölçüsel tamlığı olumsuz yönde etkileyecektir. Yapılan çalışmada bu ilkelerden yola çıkılarak takım yolu optimize edilmiştir (Şekil 2.1.b). Keskin dönüşler yerine süreklilik arz eden spiral takım yolları oluşturulmuştur. Böylece kesiciye gelen kesme kuvveti dar bir aralığa hapsedilmiştir.



Şekil 2.1. Wang ve Stori'nin yaptıkları çalışma a) Klasik takım yolu, b) Optimize edilmiş spiral takım yolu

Erdim, Lazoğlu ve Öztürk yaptıkları çalışmada, karmaşık 3B yüzeylerin işlenmesinde hacimsel ve kuvvet tabanlı ilerleme optimizasyonunu ele almışlardır [4]. Yapılan bu çalışmada ilerleme optimizasyon yöntemleri karşılaştırılmıştır. Hacimsel tabanlı ve kesme kuvveti tabanlı ilerleme optimizasyonu mekanizmaları irdelenerek avantajları ve dezavantajları araştırılmıştır (Şekil 2.2).

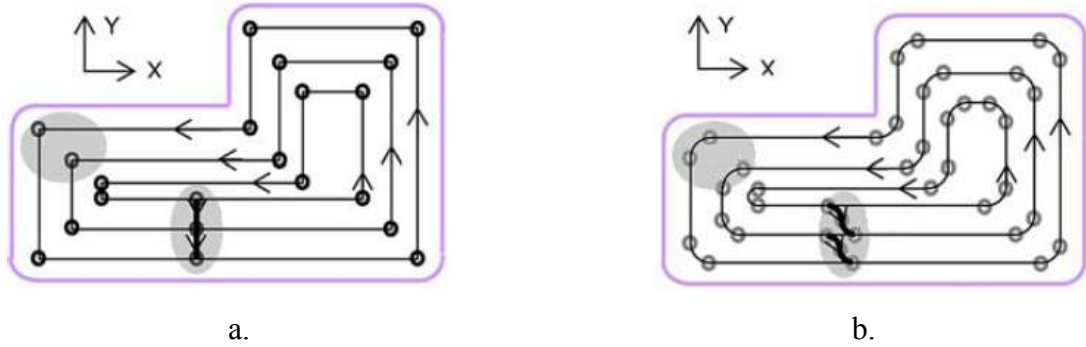


Şekil 2.2. İlerleme optimizasyonu stratejilerinin karşılaştırılması

Hacimsel tabanlı optimizasyon yönteminde hesaplamalar daha kolay yapılmaktadır. Kuvvet tabanlı optimizasyonda ise sonuçlar daha iyi çıkmaktadır.

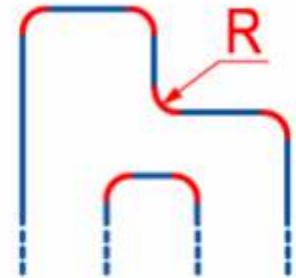
Pateloup, Duc ve Ray yaptıkları çalışmada yüksek hızlı işlemlerde, cep işleme takım yollarında köşe optimizasyonunu ele almışlardır [5]. Yüksek hızlı işlemlerde keskin dönüşler hem işleme merkezi için hem de işlenen parça ve kesici için birçok olumsuzluklar getirmektedir. Bu olumsuzluklardan en önemlisi yüksek hızlarda işleme merkezinde oluşacak dinamik yüklerin işleme merkezinin kinematikini bozmasıdır. Yapılan bu çalışmada klasik yöntemlerdeki (Şekil 2.3.a) keskin dönüşler BSpline eğrilerine dönüştürülerek (Şekil 2.3.b) işleme merkezi hareket eksenlerinde

ani dur kalk hareketleri ortadan kaldırılmıştır. Yapılan çalışma sonucu göstermiştir ki köşelerdeki yumuşak dönüşler sayesinde daha yüksek ilerleme değerlerine çıkılabilmektedir. Sonuç olarak işleme zamanında %25'e varan düşüşler elde edilebilmiştir.



Şekil 2.3. Pateloup, Duc ve Ray'ın yaptıkları çalışma, a) Klasik takım yolu, b) Dönüşler BSpline ile yumuşatılmış takım yolu

Pateloup, Duc ve Ray'ın yaptıkları çalışma bugün bütün ticari yüksek hızlı işleme yazılımlarında kullanılmaktadır (Şekil 2.4).

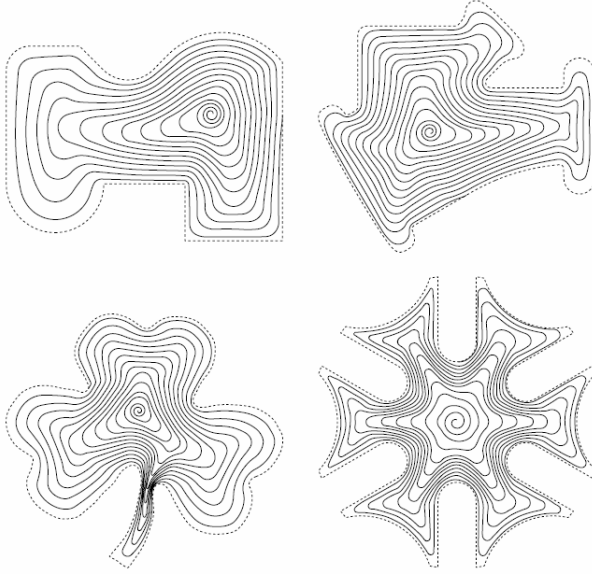


Şekil 2.4. SolidCAM yazılımında yüksek hızlı işlemede köşe yumuşatmaları

Lazoğlu, Manav ve Murtezaoğlu yaptıkları çalışmalarında karmaşık yüzeylerin işlenmesinde seçilen proses parametrelerin (ilerleme, devir sayısı, takım çapı gibi) önceden belirlenmiş maksimum kesme yüküne göre optimizasyonunu ele almışlardır [6]. Optimizasyon başlangıcında parça geometrisi ızgara şeklinde alanlara bölünerek her bölge için sekiz farklı yönde referans kuvvet haritası oluşturulmuştur. Burada

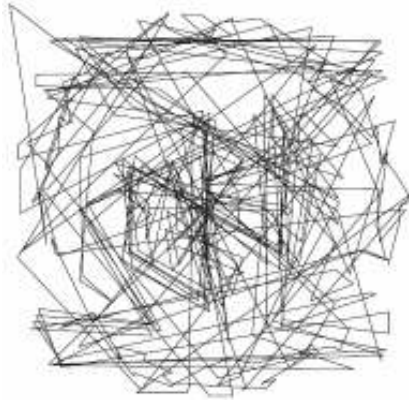
tüm noktalarda kesme kuvveti büyüklükleri sekiz komşu noktalardan aracı olarak belirlenir. Belirlenen kesme kuvvetleri içerisinde uygun kesme kuvvetine en yakın olanı belirlenir ve takım yolu o yönde yönlendirilir.

Martin ve Cristian yaptıkları çalışmada ada içermeyen 2 boyutlu ceplerin yüksek hızlı işlenmesi için yumuşak spiral takım yolunu ele almışlardır [7]. Konvansiyonel takım yolu tiplerindeki köşe dönüşlerinde meydana gelen ani ivmelenmeleri ve talaş kesiti artışlarını ortadan kaldırmak için böyle bir yöntem ele alınmıştır. Yapılan bu çalışma sonucu elde edilen bulgular, kesici ve işleme merkezi ömrünün arttığı göstermiştir. Bu çalışma sonucu elde edilen yumuşak spiral takım yolu örnekleri Şekil 2.5’de gösterilmektedir.



Şekil 2.5. Martin ve Cristian’ın yaptıkları çalışma sonucu elde edilen yumuşak spiral takım yolu örnekleri

Oysu ve Bingöl yaptıkları çalışmada melez genetik algoritma kullanarak kesici takımın boşta gezmelerini minimize etmişlerdir [8]. Yapılan bu çalışma sonucu konvansiyonel takım yoluna göre %47 oranında kesici takımın boşta gezmeleri azaltılmıştır (Şekil 2.6).



Şekil 2.6. Oysu ve Bingöl'ün yaptıkları çalışma sonucu takım boşta gezmelerinin minimuma indirilmiş takım yolu

Yapılan bu çalışmada ise, takım yolu optimizasyonunda işleme toleransı parametresinin takım yolu üzerindeki etkilerinin nasıl olacağı araştırılmıştır. CAM programlarında oluşturulan takım yollarının işleme toleransı değiştikçe nasıl etkilendikleri incelenmiştir. İşleme toleransından küçük takım yollarının kaba işlemlerde çıkarılmasıyla takım yolunda optimizasyona gidilerek varolan takım yolu satır sayısının azaltılması sağlanmıştır. Optimize edilmiş yeni takım yolu ile daha verimli bir üretimin gerçekleştirildiği öngörülmüştür.

3. CNC TAKIM TEZGAHLARI

Bilgisayarlı Sayısal Denetim (Computer Numerical Control), işlemlerin sayı, harf vb. sembollerden meydana gelen ve belirli bir mantığa göre kodlanmış komutlar yardımıyla yapılmasıdır. İmalatta bu sistemi kullanan tezgahlara CNC tezgahları denilmektedir.

Bilgisayarlı Sayısal Denetim’de tezgah kontrol ünitesinde imalat programlarının muhafaza edilebilmelerinin yanında parça üretiminin her aşamasında programı durdurma, programda gerekli olabilecek değişiklikleri yapabilme, programa kalınan yerden tekrar devam edebilme mümkündür. Bu nedenle programın kontrol ünitesine bir kez yüklenmesi yeterlidir.

CNC takım tezgahları, metal ve diğer tür malzemelerin talaş kaldırmak suretiyle işlenmesinde kullanılan her türlü takım tezgahında yaygın olarak uygulanmaktadır. Bu tezgahlardan bazıları şunlardır:

- Torna tezgahı (Lathe Machine)
- Freze tezgahı(Milling Machine)
- Matkap tezgahı (Drilling Machine)
- Delik Büyütme Tezgahı (Boring Machine)
- Taşlama Tezgahı (Grinding Machine)

Her sistemde olduğu gibi CNC tezgah ve sistemlerinin avantajları ve dezavantajları mevcuttur. Aşağıda bu avantaj ve dezavantajlar sıralanmıştır.

CNC takım tezgahlarının avantajları

- Konvansiyonel tezgahlarda kullanılan bazı bağlama kalıp, master vb. elemanlarla kıyaslandığı zaman tezgahın ayarlama zamanı çok kısadır.
- Ayarlama, ölçü, kontrolü, elle hareket vb. nedenlerle oluşan zaman kayıpları ortadan kalkmıştır.

- İnsan faktörünün imalatla fazla etkili olmamasından dolayı seri ve hassas imalat mümkündür.
- Kalifiye insan ihtiyacına gerek yoktur.
- Tezgah operasyonları yüksek bir hassasiyete sahiptir.
- Tezgahın çalışma temposu her zaman yüksek ve aynıdır.
- Her türlü sarfiyat (elektrik, emek, malzeme vb.) asgariye indirgenmiştir.
- İmalatta operatörden kaynaklanacak her türlü kişisel hatalar ortadan kalkmıştır.
- Kalıp, master, şablon vb. pahalı elemanlardan faydalanılmadığı için sistem daha ucuzdur.
- Depolamada daha az yere gerek vardır.
- Parça imalatına geçiş daha süratlidir.
- Parça üzerinde yapılacak değişiklikler sadece programın ilgili bölümünde ve tamamı değiştirilmeden seri olarak yapılır. Bu nedenle CNC takım tezgahlarıyla yapılan imalat büyük bir esnekliğe sahiptir

CNC takım tezgahlarının dezavantajları

Her sistemde olduğu gibi CNC tezgah ve sistemlerinin avantajları yanında bazı dezavantajları mevcuttur. Bunlar şunlardır;

- Detaylı bir imalat planı gereklidir.
- Pahalı bir yatırımı gerektirir.
- Tezgahın saat ücreti yüksektir.
- Konvensiyonel tezgahlara kıyaslandığında daha titiz kullanım ve bakım isterler.
- Kesme hızları yüksek ve kaliteli kesicilerin kullanılması gerekir.
- Peryodik bakımları uzman ve yetkili kişiler tarafından düzenli olarak yapılmalıdır.

Günümüzde kullanılan CNC takım tezgahlarına bir örnek Şekil 3.1’de gösterilmektedir.



Şekil 3.1. CNC takım tezgahı

3.1. Kontrol Üniteleri

CNC tezgahlarında değişik kontrol ünitesi birimleri kullanılır. Bazı kontrol ünitesi üreticileri şu şekildedir;

- FANUC,
- SIEMENS,
- BOSCH,
- MAZATROL,
- OKUMA,
- HEIDENHAIN,
- MITSUBISHI,

CNC tezgahlarında farklı kontrol üniteleri kullanılmasına karşın sayısal denetim kodları ve veri yapısı DIN 66025 ve ISO 6983 ile standart hale getirilmiştir. Kontrol üniteleri arasında bazı farklılıklar olabilmektedir. Bu farklılıklar genellikle çevrimlerdedir. Şekil 3.2’de Sayısal denetim kontrol ünitesi gösterilmektedir.



Şekil 3.2. Sayısal denetim kontrol ünitesi

3.2. Sayısal Denetim Veri Yapısı

Sayısal denetim, tezgahta işlenmek amacıyla yazılan bir işleme merkezi programı kontrol sistemine gerekli olan kontrol bilgilerini sağlayabilmek için önceden tanımlı bazı formatlara uymak zorundadır. Genel olarak CNC kontrol sistemlerinde kelime adres formatı kullanılmaktadır. Çizelgel 3.1’de örnek bir program satırı verilmiştir.

Çizelge 3.1. Örnek bir CNC program

Program kodları	İçerik
O10000;	- Program numarası ve adı,
N102 G21;	- Başlangıç bölümü ve komutları,
N104 G28 G91 X0 Y0 Z0;	
N106 G0 G17 G40 G49 G80 G90;	
N108 T2 M6;	
N110 G0 G90 G54 X30. Y40.;	
N112 G43 H2 Z50.;	
N114 M03 S800;	- Devir sayısı ve ilerleme hızı bölümü,
N116 G0 Z5. M08;	
N118 G1 Z-25. F800;	
N120 G42 D15 Y0.0;	- Ana program bölümü ve komutları,
N122 X446. F150;	
N124 Y90.;	
N126 X0.0;	
N128 Y-20.;	
N130 G1 F1000 Y-30. G40;	
N132 G0 Z50. M09;	
N134 M05;	- Program sonu ve komutları,
N136 G28G91X0Y0Z0;	
N138 M30;	
%	

Kontrol sistemlerinde, makineye verilen kontrol bilgileri çok sayıda alfabetik karakter ve çok haneli sayısal karakterlerden oluşmaktadır. Her bir alfabetik karakterin sayısal denetim kontrol sisteminde bir anlamı olduğundan dolayı bunların belirli bir sıra içinde verilmesi gerekmektedir. Çizelge 3.2’de sayısal denetim kontrol ünitesi kontrol parametreleri gösterilmektedir.

Çizelge 3.2. Sayısal denetim kontrol ünitesi kontrol parametreleri

Karakter		Anlamı
A	-	X eksenine göre açısal konumlandırma,
B	-	Y eksenine göre açısal konumlandırma,
C	-	Z eksenine göre açısal konumlandırma,
D	-	İkinci takım fonksiyonu,
E	-	İkinci ilerleme fonksiyonu,
F	-	İlerleme fonksiyonu,
G	-	Hazırlık fonksiyonları (G00 ... G99)
I	-	X eksenine paralel interpolasyon değeri,
J	-	Y eksenine paralel interpolasyon değeri,
K	-	Z eksenine paralel interpolasyon değeri,
M	-	Yardımcı fonksiyonlar (M00 ... M99),
N	-	Satır numarası,
P	-	X eksenine paralel üçüncü parametre,
Q	-	Y eksenine paralel üçüncü parametre,
S	-	Devir sayısı fonksiyonu,
T	-	Takım fonksiyonu,
U	-	X eksenine paralel ikinci parametre,
V	-	Y eksenine paralel ikinci parametre,
W	-	Z eksenine paralel ikinci parametre,
X	-	X ekseninde mesafe,

G ve M parametreleri yan taraflarına yazılan rakama göre anlam ifade etmektedirler. DIN 66025 ve ISO 6983 standardına göre G ve M kodlarına standart getirilmiştir. İşleme merkezi kontrol ünitelerinin büyük bir çoğunluğu bu kodları kullanmaktadır.

Bazı kontrol ünitelerinde ufak değişiklikler olabilmektedir. Bazı kontrol üniteleri ise tamamen kendine özgü kod sistemi kullanabilmektedir. G hazırlık fonksiyonları ve açıklamaları Çizelge 3.3’de, M yardımcı fonksiyonları ve açıklamaları Çizelge 3.4’de gösterilmektedir.

Çizelge 3.3. Bazı G kodları (Fanuc kontrol sistemine göre) ve açıklamaları

G kodu		Anlamı
G00	-	Talaş kaldırmadan hızlı ilerleme,
G01	-	Talaş kaldırarak belirtilen F ilerleme değerinde ilerleme,
G02	-	Saat yönünde talaş kaldırarak ilerleme,
G03	-	Saat ters yönde talaş kaldırarak ilerleme,

Çizelge 3.4. Bazı M yardımcı fonksiyonları ve açıklamaları

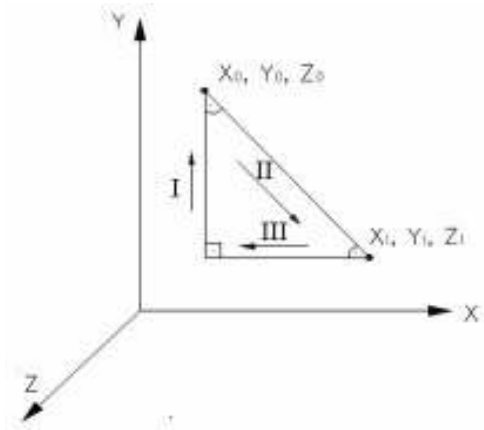
M kodu		Anlamı
M00	-	Program durdurma,
M01	-	Opsiyonel durdurma,
M02	-	Program sonu,
M05	-	İş / kesici milin durması,
M06	-	Takım değiştirme,

3.3. Hareket Tipleri

CNC tezgahlarında kesici hareketlerinin yapılmasında değişik hareket tipleri kullanılmaktadır. Bu hareketler, temelde iki tip hareketten oluşmaktadır.

3.3.1. Doğrusal hareket

İşleme merkezlerinde kesiciyi bir noktadan diğer bir noktaya doğrusal olarak konumlandırmak için G00 ve G01 kodları kullanılmaktadır (Şekil 3.3).



Şekil 3.3. Doğrusal hareketler

Şekil 3.3’de görüldüğü gibi II nolu hareket belirli bir açı altında yapıldığından bu harekete açısal hareket (interpolasyon) denilmektedir.

G00 Kodu

Kesiciyi talaş kaldırmadan konumlandırır. İlerleme değeri olarak işleme merkezinin boşta ilerleme değeri alınır. Kesici G00 hareketini yaparken kesinlikle parçadan talaş kaldıramaz. Bu kod G00 şeklinde yazılabileceği gibi G0 şeklinde de yazılabilmektedir. Aşağıda G0 kodu için bir satır gösterilmektedir.

G0 Z103.2

Fanuc kontrol ünitesine sahip tezgahlarda, uzunluk belirten değerlerin sonunda nokta işareti vardır. Bu nokta işareti olmaz ise ölçü mikrometre kabul edilmektedir.

G01 Kodu

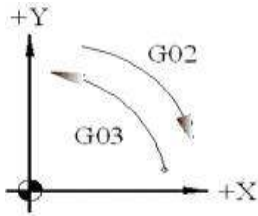
Kullanımı G00 ile aynıdır. G01 şeklinde yazılabileceği gibi G1 şeklinde de yazılabilmektedir. G1 fonksiyonunda kesici F ilerleme fonksiyonu ile belirtilen hızda hareket etmektedir. Aşağıda G1 kodu içeren satır gösterilmektedir.

G1 Z97.143 F300.

Kesici G1 hareketi ile talaş kaldırarak ilerlemektedir. İlerleme değeri 300 mm/dak'dır.

3.3.2. Dairesel hareket

Dairesel talaş kaldırarak ilerleme için kullanılır [9]. G02 ve G03 şeklinde yazılabileceği gibi G2 ve G3 şeklinde de yazılabilmektedir. Bu tür hareketler daireysel interpolasyon hareketi olarak da adlandırılmaktadır. Dairesel interpolasyon Şekil 3.4'de gösterilmektedir.



Şekil 3.4. G2 ve G3 daireysel interpolasyonu

G2 ve G3 interpolasyonunun kullanım formatı şu şekildedir.

N_ G02 X_ Y_ R_ F_ ;

veya

N_ G02 X_ Y_ I_ J_ F_ ;

Buradaki değişkenlerin anlamları şöyledir.

X - Yayın bitiş noktasında kesici merkezinin X koordinatları,

Y - Yayın bitiş noktasında kesici merkezinin Y koordinatları,

R - Yayı elde etmek için çakı merkezinin çizmiş olduğu yayın yarıçapı,

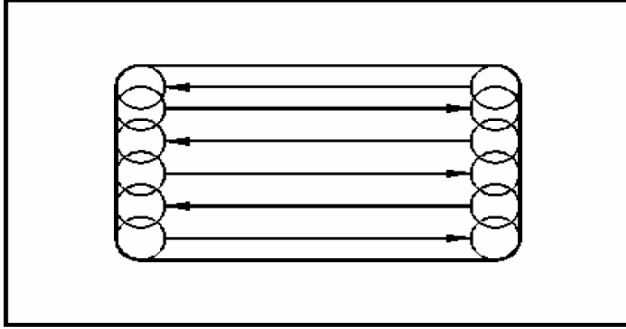
I - Yay başlangıç noktasında çakı merkezinin, yay merkezine X ekseninde artışı olan uzaklığı,

J - Yay başlangıç noktasında çakı merkezinin, yay merkezine Y ekseninde artışı olan uzaklığı,

F - Hareket esnasında ilerleme hızı,

3.4. Çevrimler

Sayısal denetim işleme merkezlerinde çevrim; bir dizi işlemi gerçekleştirmek için kendi bünyesinde bulunan komut sistemini barındıran ve bunu kullanıcının tek bir komutla gerçekleştirebildiği işlemler dizisidir (Şekil 3.5).



Şekil 3.5. Örnek çevrim.

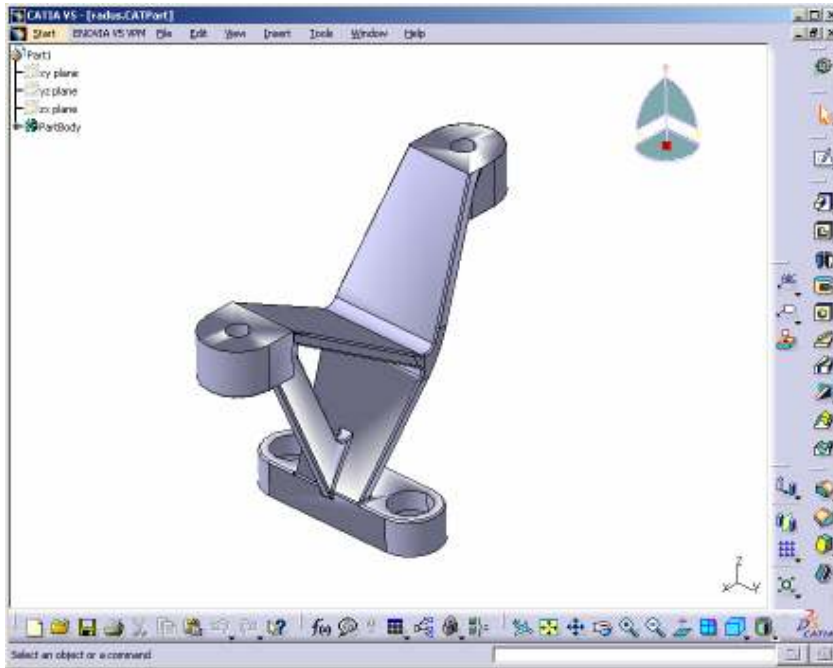
Çevrimlerin tekrar sayısı çevrim bloğundaki K değeri ile belirtilir. K değeri mutlak mod (G90) fonksiyonu aktif halde iken çevrimde belirtildiğinde çevrim aynı pozisyonda K defa tekrarlanır. Tekrar sayısı çevrimde belirtilmediği zaman işlemin bir defa yapılacağı anlaşılır.

4. CAD/CAM

Gelişen teknolojiyle birlikte, bilgisayar destekli tasarım (CAD) ve bilgisayar destekli imalat (CAM) üretimin her alanına hızla girmiştir. CAD CAM verileri, imalatta kolaylık, zamandan ve iş yükünden tasarruf, kalitede artış, üretimde geniş bir yelpaze seçeneği ve bunun gibi birçok avantajı beraberinde getirmiştir.

4.1. Bilgisayar Destekli Tasarım (CAD)

Makine imalat sektöründe BDT, 2 veya 3 boyutlu parçaların modellenmesinde ve imalat için gerekli çıktıların oluşturulmasında kullanılır. Sektörün gelişmesi ile birlikte birçok BDT yazılımı ticari olarak kullanıma sunulmaktadır. Örnek olarak CATIA, Siemens NX, Pro/Engineer, SolidWorks, MasterCAM, vb. verilebilir. Şekil 4.1'de CATIA ile yapılan bir tasarım görülmektedir.

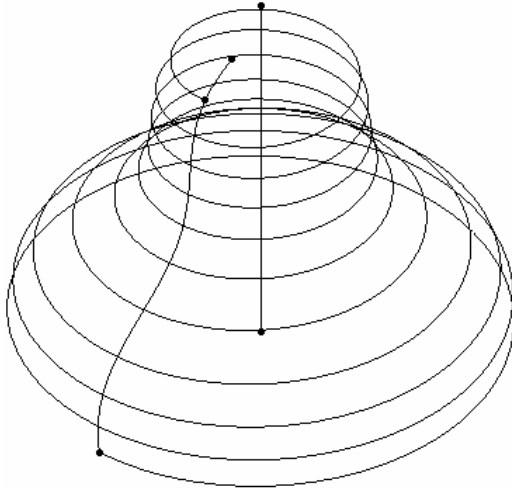


Şekil 4.1. CATIA ile yapılan bir tasarım

BDT yazılımları 3B tasarımların oluşturulmasında 3 temel yöntemi esas alırlar. Bunlar sırası ile şu şekildedir.

Tel Kafes Model

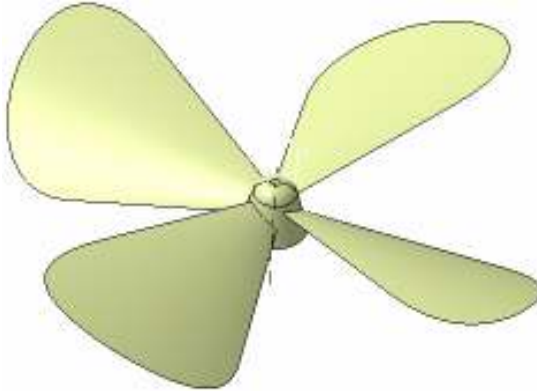
Tel kafes tasarım modelin çizgilerden oluştuğu modelleme yöntemidir. Çizgiler 3 boyutlu olabildiği gibi 2 boyutlu da olabilmektedir. Modellemeye ilk başlangıç genellikle 2 ve 3 boyutlu tel kafes modellerle başlanır. Şekil 4.2’de örnek tel kafes model verilmiştir.



Şekil 4.2. Tel kafes model

Yüzey Model

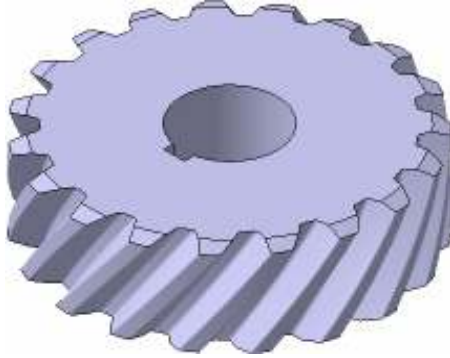
BDT’da 2 veya 3 boyutlu uzayda alan tanımlayan ve hacmi olmayan nesneleri yüzey model denir. Yüzey modellerin oluşturulmasında 2 ve 3 boyutlu tel kafes modellerden yararlanılır. Şekil 4.3’de yüzey model örneği verilmiştir.



Şekil 4.3. Yüzey model örneği

Katı Model

BDT’da 3 boyutlu uzayda hacim kaplayan nesnelere katı model denir. Katı modellerin oluşturulmasında 2 ve 3 boyutlu tel kafes ve yüzey modeller kullanılır. Şekil 4.4’de katı model örneği verilmiştir.



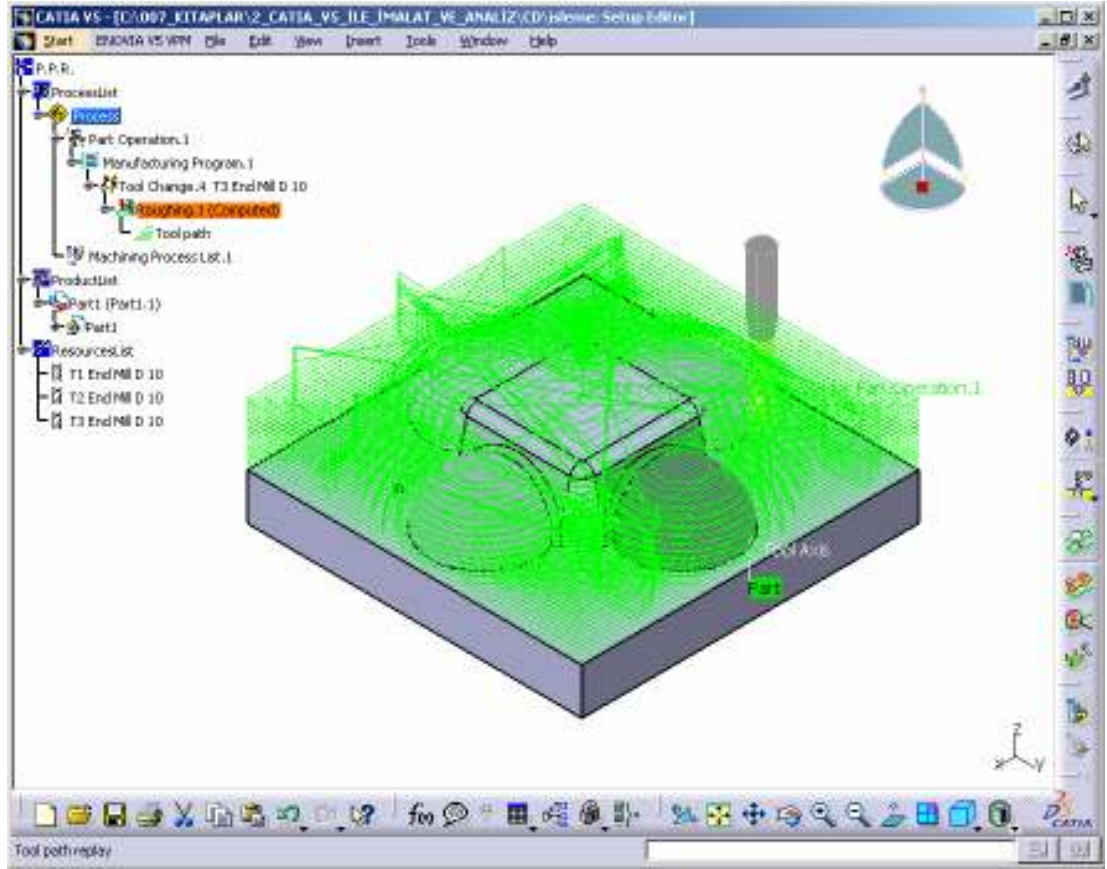
Şekil 4.4. Katı model örneği

4.2. Bilgisayar Destekli İmalat (CAM)

Bilgisayar Destekli İmalat (Computer Aided Manufacturing), BDT sürecinde elde edilen ürünü alıp imalat için gerekli çıktıların oluşturulması için kullanılan bir araçtır. CAM alanında birçok ticari yazılım geliştirilmiştir. Örnek olarak:

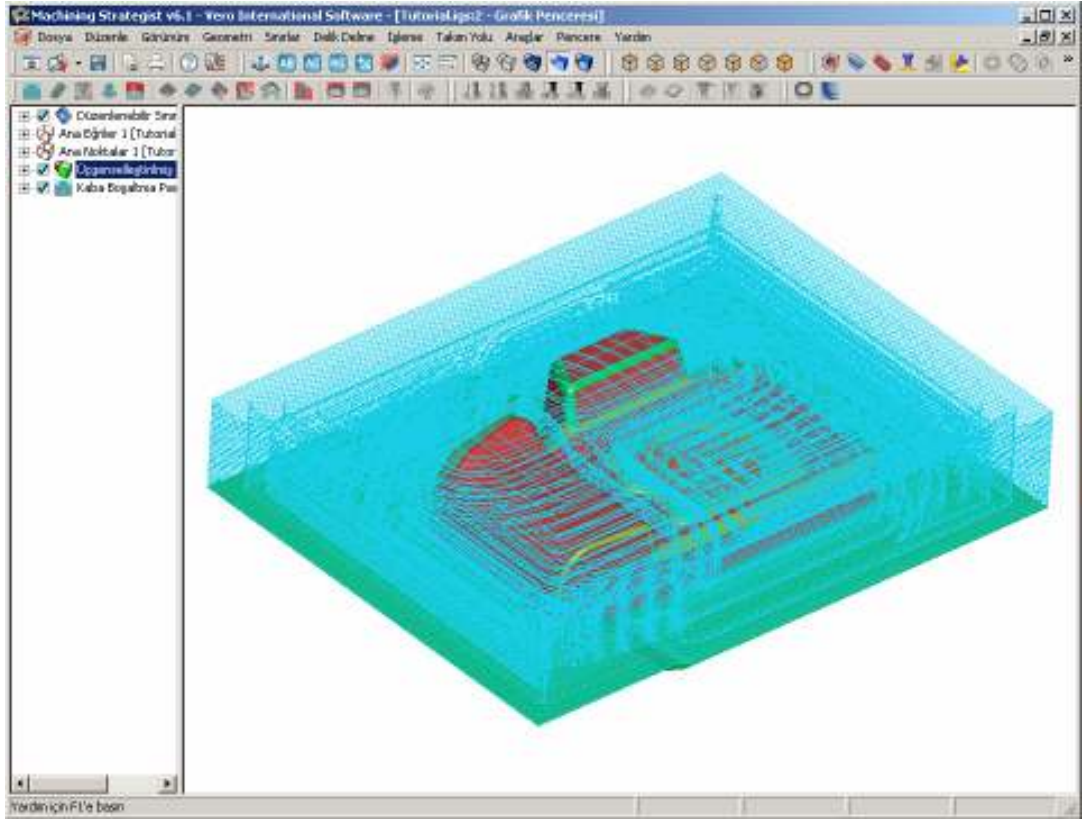
- MasterCAM
- CATIA
- Siemens NX
- Machining Strategy
- Auton

Şekil 4.5’de CATIA V5 yazılımında bilgisayar destekli imalat modülü görülmektedir.



Şekil 4.5. CATIA V5 yazılımında bilgisayar destekli imalat

MasterCAM, CATIA, Siemens NX yazılımları BDT ve BDİ modüllerini bütünleşik olarak tek ara yüzde sunmaktadır. Bunun yanı sıra Machining Strategy, Auton ve Esprit gibi yazılımlarda sadece takım yolu türeten yazılımlardır. Şekil 4.6'da Machining Strategy yazılımında takım yolu örneği verilmiştir.



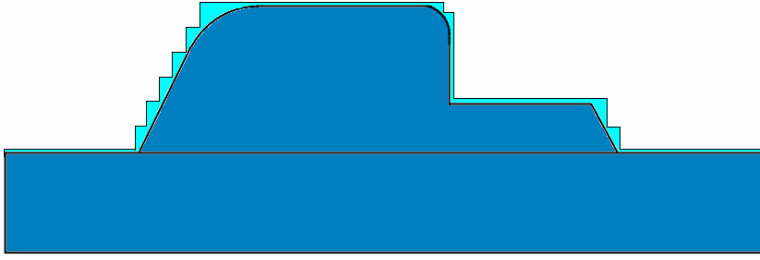
Şekil 4.6. Machining Strategy yazılımında bilgisayar destekli imalat

4.2.1. Bilgisayar destekli imalat basamakları

Bilgisayar destekli imalatın yapılmasında başlıca üç bulunmaktadır.

1. Kaba işleme operasyonları:

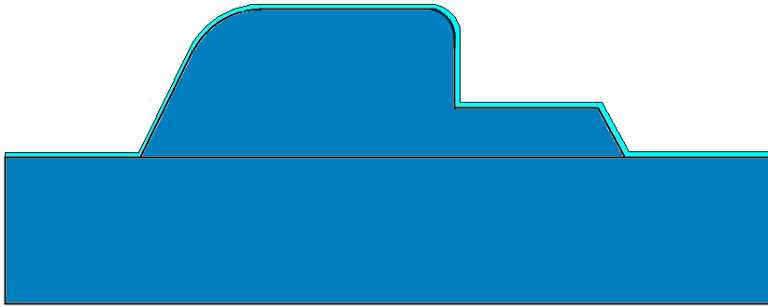
Kaba işleme operasyonunda hedef minimum zamanda maksimum kütleyi işlemektir. Yüzey kalitesi ve ölçüsel tamamlık önemli değildir. Fakat, kesici takım ömrü maliyeti etkileyeceğinden önemlidir. BDİ operasyonlarının en çok zaman alan adımı olduğu için işleme zamanının düşürülmesi oldukça önemlidir. Kaba işleme operasyonlarında kullanılacak kesici, bu beklentileri karşılayacak nitelikte olmalıdır. Ayrıca, işleme parametreleri de kaba işleme de verimliliği arttıran önemli bir parametre grubudur. Şekil 4.7’de kaba işleme sonrası parça durumu görülmektedir. Şekilden de görüldüğü gibi parça geometrisi ile işleme sonrası oluşan geometri arasında düzensiz bir kesit dağılımı söz konusudur.



Şekil 4.7. Kaba işleme operasyonu

2. Ara işleme operasyonları

Kaba işleme sonrası parçanın her yerinde düzensiz talaş kalır. Bu şekilde iken bitirme operasyonu uygulanamaz. Bu yüzden kaba işleme ile bitirme işleme operasyonları arasına ara operasyonlar eklenmelidir. Bu operasyonlarda hedef parçanın her yerinde eşit miktarda bitirme payı bırakmaktır. Böylece bitirme operasyonunda istenen yüzey kalitesi ve boyutsal tolerans elde edilmiş olur. Şekil 4.8’de kaba işlemenin ardından gerçekleştirilen ara operasyonlar sonucu, parçanın her yerinde bitirme payının eşitlenmesi görülmektedir.

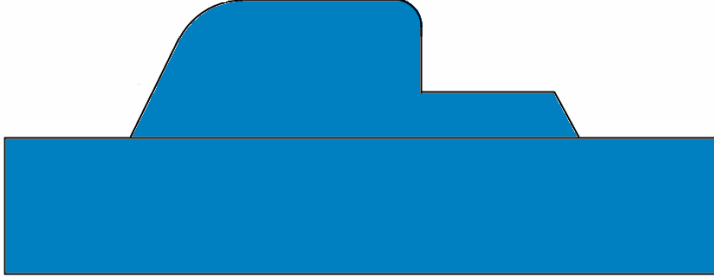


Şekil 4.8. Ara işleme operasyonu

3. Bitirme işleme operasyonları

Ara işleme operasyonları ile parçanın her yerinde bitirme payı eşitlendikten sonra bitirme operasyonları uygulanır. Bitirme operasyonu tek adımda gerçekleştirilebileceği gibi birkaç basamakta da gerçekleştirilebilir. Bitirme işleme operasyonlarında hedef en iyi yüzey kalitesidir. Bu yüzden kesici seçimi oldukça önemlidir. Kesme ve işleme parametreleri de yüzey kalitesi üzerinde etkilidirler.

Şekil 4. 9’da bitirme operasyonunun ardından parça geometrisi görülmektedir. Parça orijinal ölçüsüne getirilmiştir.

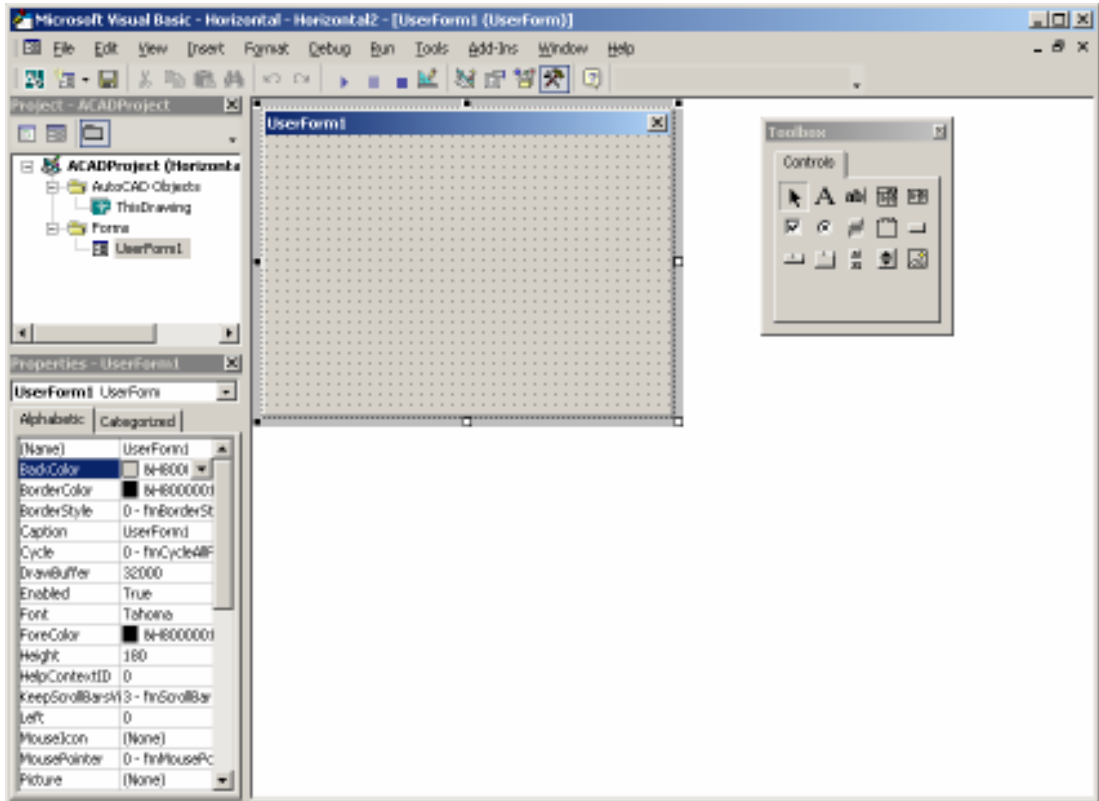


Şekil 4.9. Bitirme işleme operasyonu

5. VISUAL BASIC VE VISUAL BASIC APPLICATION

5.1. Visual Basic (VB)

Microsoft firması tarafından geliştirilen Visual Basic yazılımı (Şekil 5.1), QBASIC derleyicisinin geliştirilmiş ve Windows ortamına uyarlanmış üst seviye programlama dilidir. Windows ortamına uyarlandığı için nesneye yönelik bir dildir. VBX kontrollerini destekleyen ilk dillerden biridir. Visual Basic’de, 1.0 sürümünden 6.0 sürümüne kadar bir çok yenilik ve değişiklik olmuştur. Bunlardan biri de, arayüzünün güçlü ve etkili bir görünüm kazanmasıdır. Visual Basic, devamlı geliştiği bu süre sonunda yüksek hızlı uygulamalar, OLE sunucular, ActiveX kontrolleri ve daha bir çok uygulamayı geliştirilebilecek hale gelmiştir.



Şekil 5.1. Visual Basic 5 yazılımının genel görünümü

Microsoft Windows için program geliştiren programcıların %25 Visual Basic'i tercih etmektedirler. Visual Basic'i en popüler programlama dillerinden biri yapan en önemli nedenlerden biri de kullanımı kolay olmasıdır. Visual Basic de program

yazmak için çok fazla teknik yazılım bilgisine sahip olmak gerekmemektedir. Kontrolleri form üzerine yerleştirmek ve kodu yazmak yeterlidir.

Visual programlama dillerinin en önemli özelliklerinden biri programın ekran tasarımını kodlama ile değil tasarım zamanı bir resim çiziyormuş gibi rahatça yapılabilmesidir [9].

Çizelge 5.1 ‘de Visual Basic programında, öğrencinin aldığı not aralığına uygun bir mesaj veren program örneği verilmektedir.

Çizelge 5.1. öğrencinin aldığı not aralığına uygun bir mesaj veren program örneği

```
Private Sub UserForm_Click()
    Dim v
    v = InputBox("Not girişı", "Vize notunuz:", "50")
    Select Case v
        Case 0: MsgBox ("Hiçbirşey öğrenememişsiniz.")
        Case 1 To 20: MsgBox ("Durumunuz hiç iyi değil")
        Case 21 To 40: MsgBox ("durumunuz İyi değil")
        Case 41 To 49: MsgBox (" Daha Çok çalışmalısınız")
        Case 50: MsgBox ("Sınırdan bir not aldınız")
        Case 51 To 65: MsgBox (" İyi bir not aldınız")
        Case 66 To 75: MsgBox (" Tebrikler.İyi bir not aldınız.")
        Case 76 To 100: MsgBox ("Tebrikler.Çok iyi bir not aldınız.")
        Case Else: MsgBox ("Böyle bir not almış olamazsınız.")
    End Select
End Sub
```

5.2. Visual Basic Application (VBA)

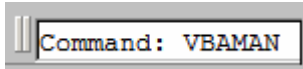
AutoCAD ortamında VBA programı; verilen bir işlevi yerine getirmek için kodların, VB sınıf birimlerinin ve AutoCAD şekillerinin birlikte kullanımı olarak tanımlanabilir

[10]. VBA projeleri AutoCAD çizimi sınırları içerisinde veya farklı dosyalar halinde kaydedilebilirler. Yerleştirilmiş projeler bir AutoCAD çizimi sınırları içerisinde kaydedilirler. Bu projelerin bulunduğu bir çizim AutoCAD ortamında açıldığında projeler otomatik olarak yüklenirler.

5.2.1. VBA yöneticisi ile projeleri düzenleme

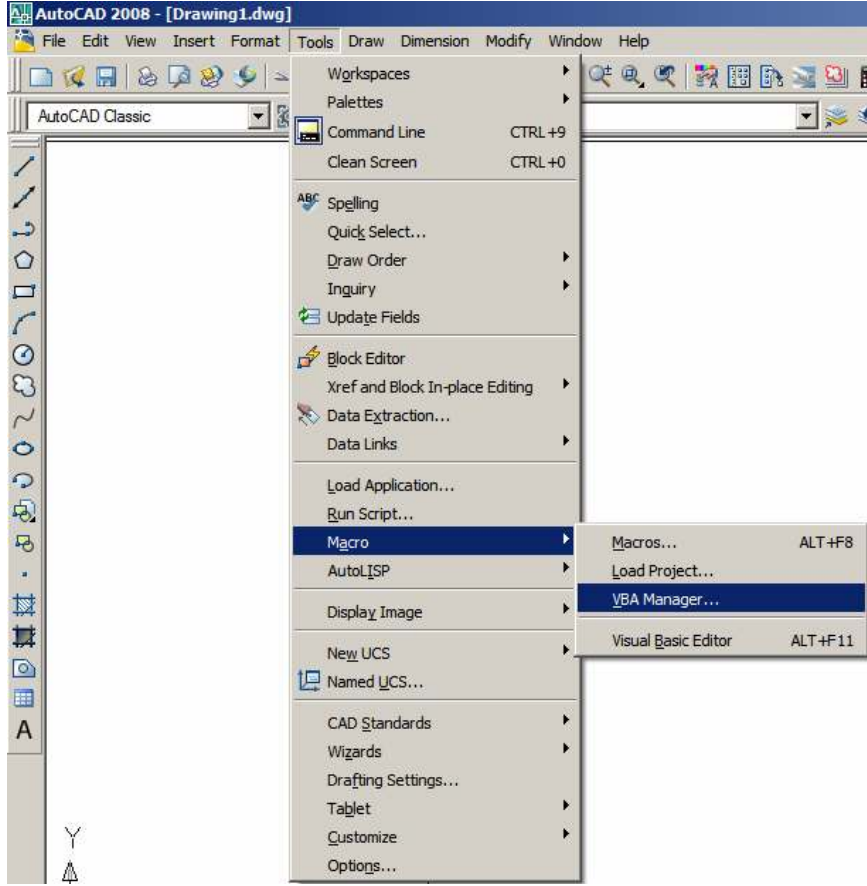
VBA yöneticisi kullanılarak geçerli AutoCAD oturumunda yüklü olan VBA projelerine bakılabilir. VBA yöneticisi; VBA projelerini yüklemeye, boşaltmaya, kaydetmeye, oluşturmaya, yerleştirmeye ve seçip çıkarmaya imkan sağlayan bir AutoCAD aracıdır.

VBA yöneticisini açmak için; AutoCAD komut satırına “VBAMAN” yazılarak onay verilmelidir (Şekil 5.2).



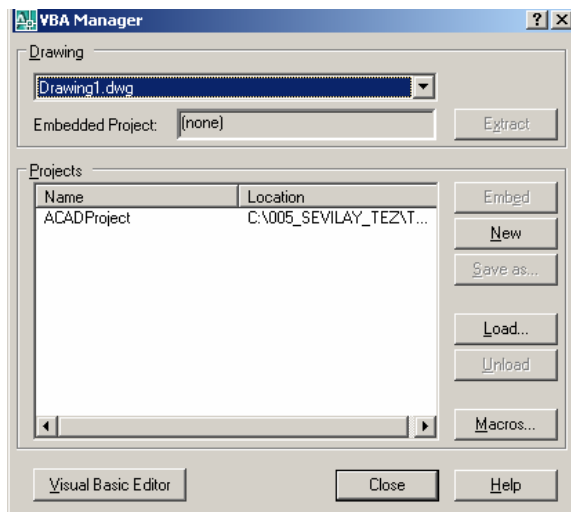
Şekil 5.2 Komut satırına “VBAMAN” yazılımı

Böylelikle VBA Manager penceresi açılacaktır. Aynı işlem AutoCAD düşey menüsündeki “Tools->Macro->VBA Manager“ seçilerek de sağlanabilir. VBA Manager komutu ile Şekil 5.3’deki gibi bir sayfa kullanıcıya sunulur.



Şekil 5.3. VBA Manager Düşey Menüsü

Açılan VBA yönetici penceresi Şekil 5.4’ de gösterilmektedir.



Şekil 5.4. VBA Manager Penceresi

VBA Manager sayfasından kullanıcı, geliştirdiği bir projeyi (programı) yükleyebilir (Load), kaldırabilir (Unload), yeni bir proje (New) başlatabilir. VBA manager sayfasında yapılabilen işlemler aşağıda sırası ile verilmiştir.

1. Bir projenin yüklenmesi (Load)

Varolan bir VBA proje dosyasını yüklemek için; VBA yöneticisindeki yükle (Load) özelliği kullanılır. Load butonuna tıklandığında DVB uzantılı dosyaları açmaya imkan tanıyan Open VBA Project Penceresi açılacaktır. Buradan istenen DVB uzantılı dosya seçilip onay verilerek Yükleme gerçekleştirilir.

2. Bir projeyi boşaltma (Unload)

Bir VBA Projesini boşaltmak için VBA yöneticisinde boşaltmak istenen proje seçilerek boşaltma (Unload) butonuna tıklanır.

3. Yeni bir proje oluşturma

Yeni projeler kaydedilmeyen ayrıntılı projeler olarak oluşturulurlar. İlk olarak bir proje oluşturulduğunda, bir çizime yerleştirilebilir veya bu proje başka bir yere, bir proje dosyası olarak kaydedilebilir. Yeni bir VBA projesini oluşturmak için VBA yöneticisi açılarak New (yeni) butonuna tıklanır.

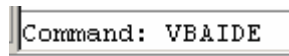
4. Projeyi kaydetme

Çizim kaydedildiği zaman, yerleştirilmiş projeler de kaydedilirler. Ayrıntılı projeler VBAIDE deki VBA yöneticisi kullanılarak kaydedilmelidir. VBA yöneticisini kullanarak projenin kaydedilmesi için VBA yöneticisi açılarak kaydedilecek proje seçilir. Save As (Farklı Kaydet) seçilerek Farklı Kaydet diyalog kutusuna ulaşılır. Kaydedilecek proje için dosya ismi girilerek Save (Kaydet) işaretlenir.

5. VBA yöneticisinin AutoCAD ortamında açılması

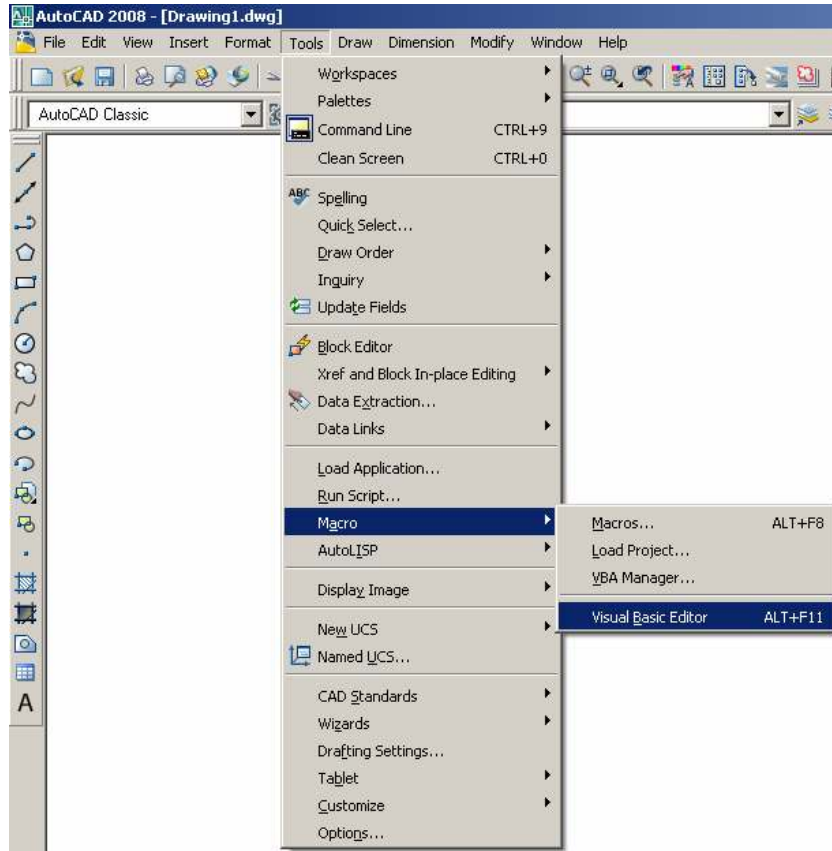
VBA editörünün AutoCAD ortamında açılabilmesi için farklı yöntemler vardır. Bunlardan birincisi;

AutoCAD komut satırına (Command), “VBAIDE” yazılarak onay verildiğinde gerçekleştirilir (Şekil 5.5).



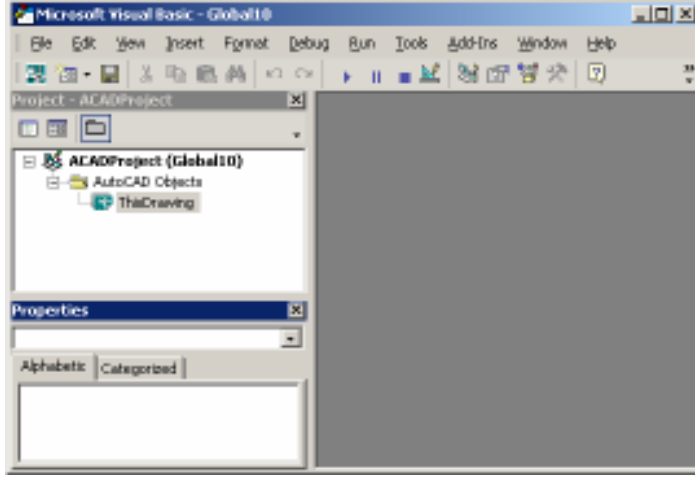
Şekil 5.5. AutoCAD Komut Satırı

İkinci yöntem; AutoCAD düşey menü çubuğundan “Tools->Makro->Visual Basic Editör” seçeneklerinin seçilmesiyle sağlanır (Şekil 5.6).



Şekil 5.6. VBA Editörüne Giriş düşey menüsü

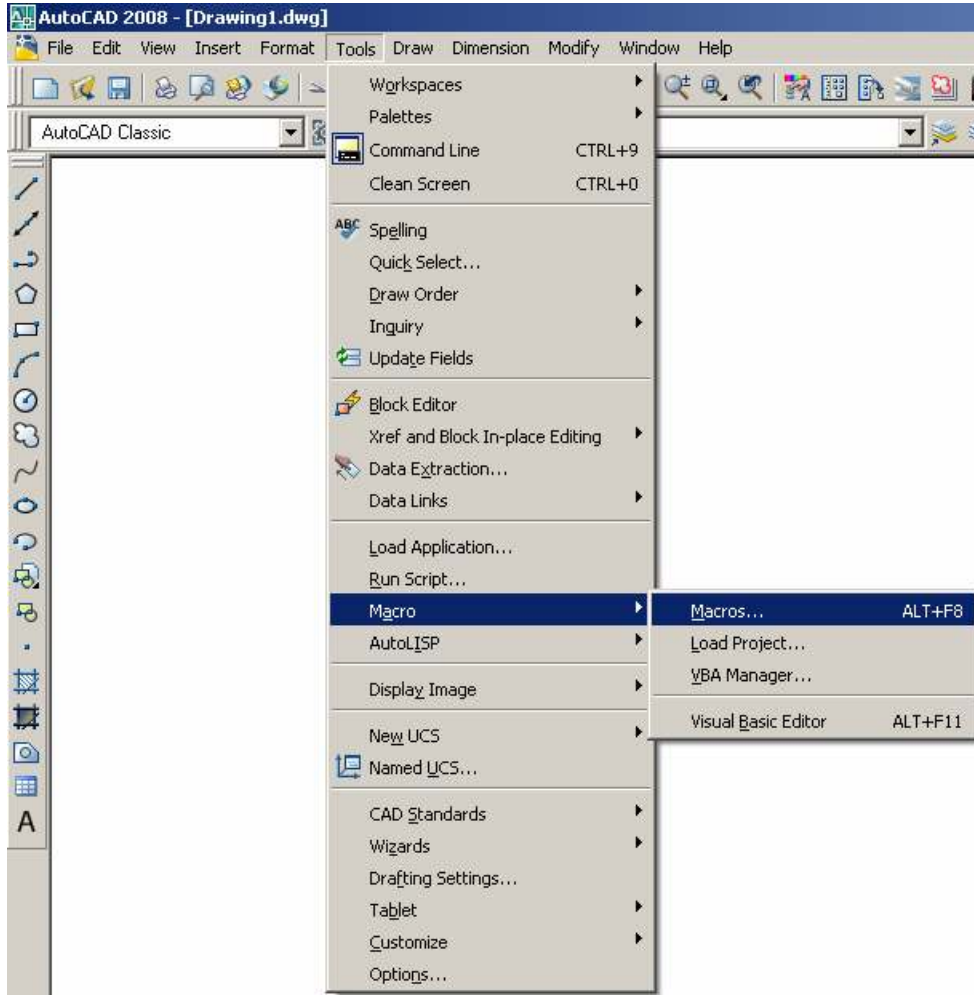
Üçüncü yöntem; “Alt+F11” tuşlarına beraber basılarak sağlanır. Böylelikle VBA editörüne geçiş sağlanır (Şekil 5.7).



Şekil 5.7. VBA Editörü

6. Macro yönetimi

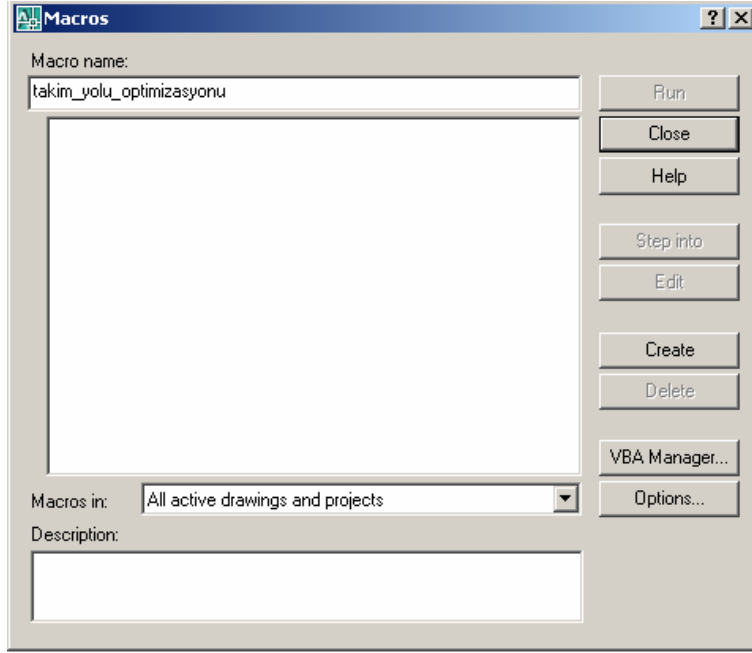
Macrolar diyalog kutusu, VBA proje seçeneklerini düzenlemeye, macroları oluşturmaya, silmeye, düzenleme ve çalıştırmaya imkan sağlarlar. Macro genel uygulanan bir alt yordamdır. Her proje çoğunlukla en azından bir macro içerir. Makro diyalog kutusunu açmak için “Tools” (Araçlar) menüsünden macro, buradan da “macros” seçeneği işaretlenir. Veya, AutoCAD komut penceresinden VBARUN komutu girilir (Şekil 5.8.)



Şekil 5.8. Macro Yönetimine Giriş düşey menüsü

7. Yeni bir macro oluşturma

Yeni bir macro oluşturmak için Makrolar (Macros) diyalog kutusu açılarak yeni bir macro için isim girilir ve Create (Oluştur) butonu seçilir. Eğer belirli bir isimde önceden varolan bir macro ismi girilirse, önceden varolan macro ile yer değiştirilip değiştirilmeyeceği sorulacaktır. Eğer gelen çağrı evet olarak seçilirse varolan macrodaki birim silinecektir ve yeni boş bir macro belirli bir isimle oluşturulacaktır. Eğer gelen çağrı iptal olarak seçilirse macro diyalog kutusu geri gönderilir ve yeni bir macro oluşturulamaz (Şekil 5.9.)



Şekil 5.9. Yeni bir macro oluşturma

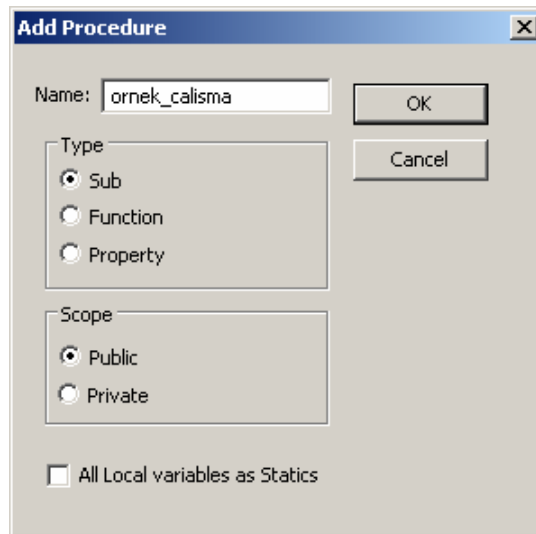
5.3. VBA’da Örnek Çalışma

Üstünde “Örnek Çalışma” yazısı yazan bir program hazırlama sırası şu şekildedir. Öncelikle Command satırına VBAIDE yazılarak VBA editörü açılır.

VBAIDE’nin içersinde bulunan “View” menüsünden “Code” opsiyonu seçilerek kod penceresi açılır.

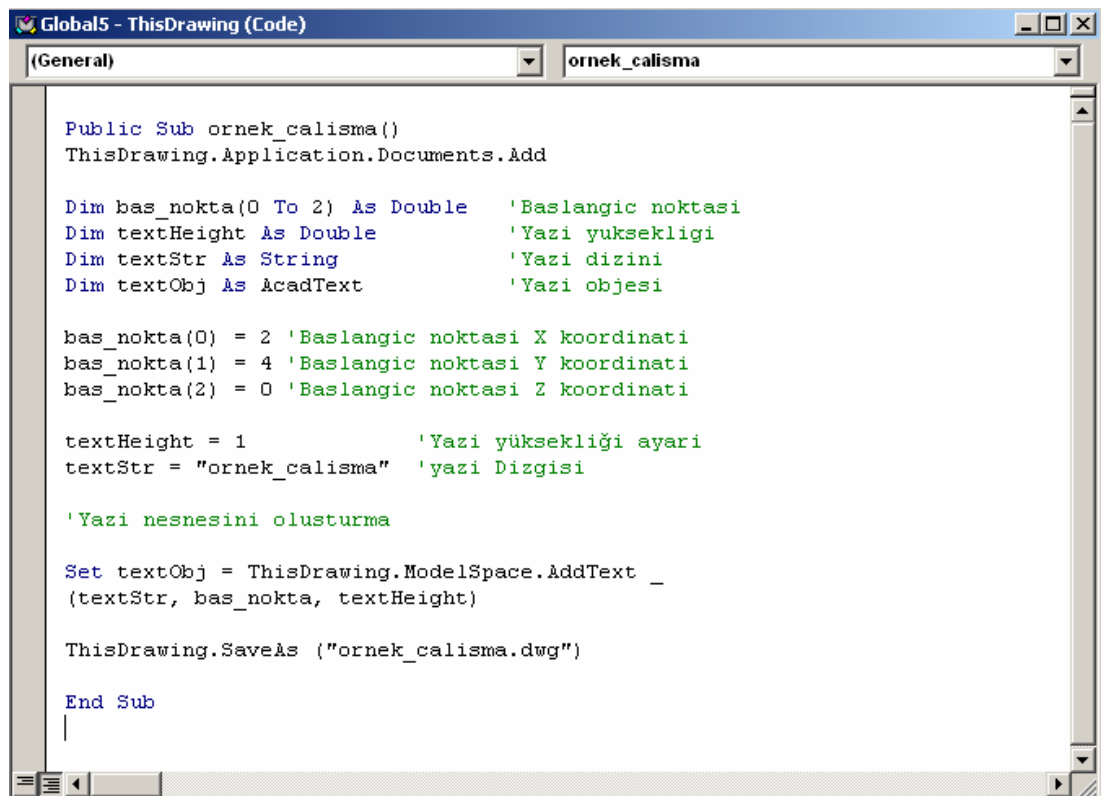
VBAIDE’nin “Insert” menüsü içerisinde bulunan “Procedure” opsiyonu seçilerek proje içersinde yeni bir prosedür oluşturulur.

Prosedür bilgilerini girerken seçilen tipin “SUB” olması ve alanının da “PUBLIC” olduğundan emin olunur. İsim olarak herhangi bir isim girilebilir. Burada “OK” butonuna basılarak işlem tamamlanır (Şekil 5.10.).



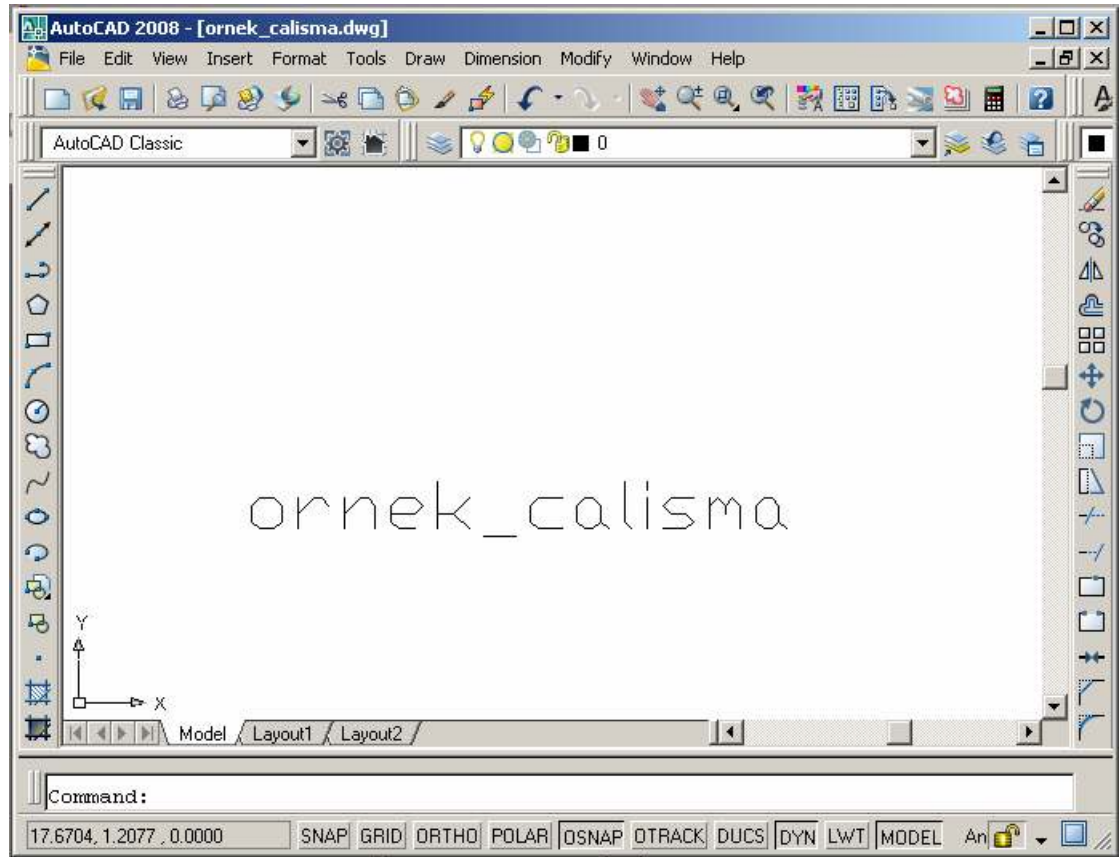
Şekil 5.10. Add Procedure penceresi

Ornek_Calisma yazabilmek için gerekli kodlar kod penceresine girilir (Şekil 5.11.).



Şekil 5.11. Kod penceresi

VBAIDE “RUN” menüsünden “Run Sub / UserForm” sekmesi ile program çalıştırıldığında belirtilen kodlara uygun olarak “ornek_calisma” yazısı ekrana yazdırılacak ve çizim ismi de “ornek_calisma” olarak düzenlenecektir (Şekil 5.12.).



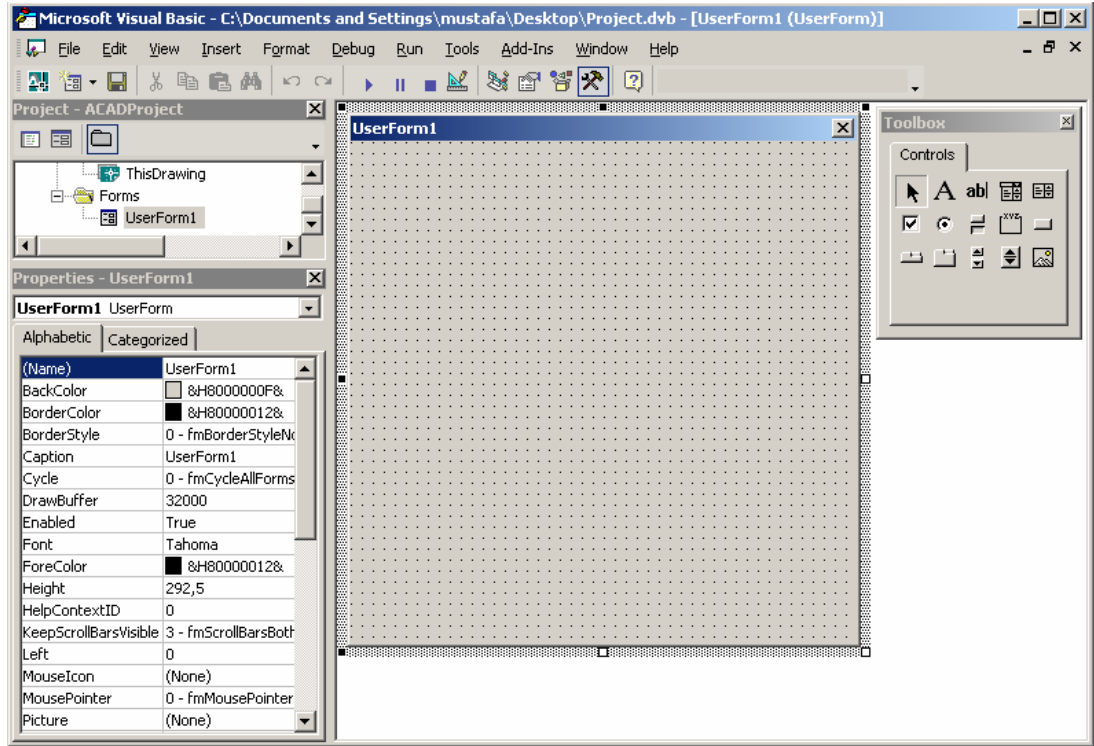
Şekil 5.12. Uygulama sonucu AutoCAD çalışma sayfası

5.4. Kullanıcı Formlarının VBA ‘da Tasarlanması

VBA dili ve VBA editörüne göre, VBA kullanıcı ortak yüz unsurunu içersinde barındırır. Bu Microsoft formları paketidir ki tüm VBA ana programlarını kapsar.

5.4.1. Çalışma sayfaları oluşturma

Yeni bir çalışma sayfası oluşturmak için Insert menüsünden Çalışma Sayfası (User Form) seçilir. Bu yeni bir sayfa oluşturarak onu tasarım modunda açar ve denetim araç kutusunu gösterir. Bu çalışma sayfası, diyalog kutusunu tasarlamak için kullanılan boş bir palettir (Şekil 5.13.).



Şekil 5.13. VBA'da yeni çalışma sayfası

5.4.2. Kontrol ekleme ve ayarlama

Çalışma sayfasını kontrol etmek için Önce, Denetim Araç Kutusunda istenen kontrole, daha sonra çalışma sayfasına tıklanır. Tüm kontroller ilk denemede istenilen yere yerleştirilmede başarılı olunmayabilir. Bir kontrolü yeni bir yere taşımak için kontrolün üzerine tıklanır, bu kontrolü aktif hale getirir. Kontrolün üzerinde bir yere tıklanarak yeni yerine taşınır.

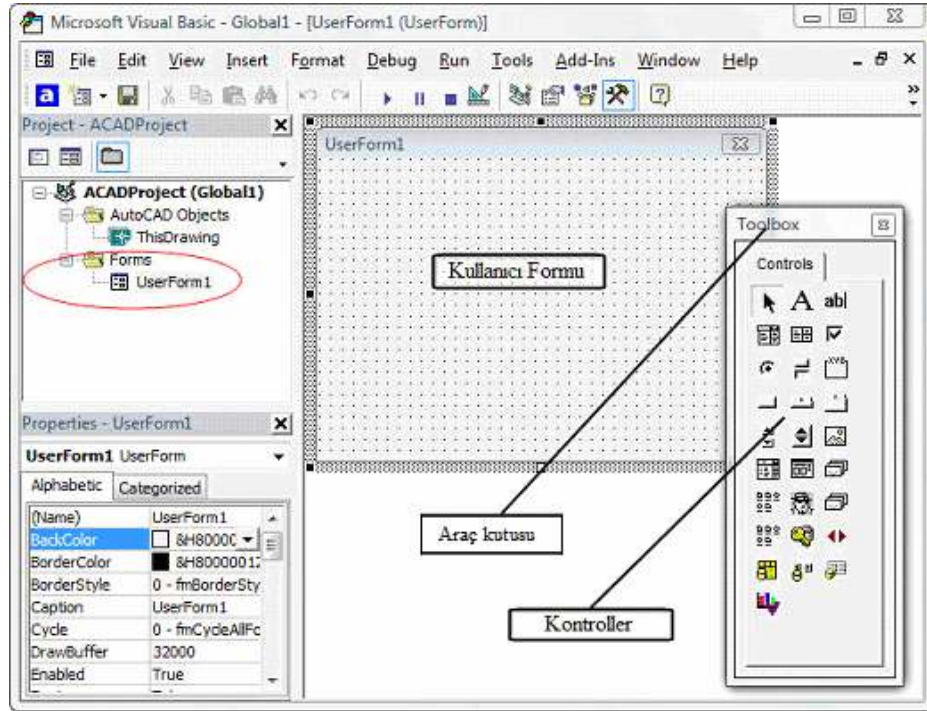
5.4.3. Özelliklerin ayarlanması

Etiket varsayılan görünüşü soluktur. Label1 diye adlandırılır ve böyle kalır. Etiket görünüşünü değiştirmek için özellikler penceresi kullanılır.

Kontroller aynı zamanda nesnelerdir. Aslında kontrolü düşünmenin tek yolu onu görünür ortak yüzlü bir nesne olarak düşünmektir.

5.4.4. Kontrol çeşitleri

VBA uygulamalarında kullanıcı ara yüz oluşturmada birçok denetim araç kutusu bulunmaktadır. Bunlardan bazıları Şekil 5.14’de toplu olarak verilmiştir.



Şekil 5.14. VBA’da kullanıcı Formu

Textbox (Metin Kutusu)

TextBox kontrolü kullanıcının çalışma sayfası üzerindeki boş kutuya metin girmesini sağlar. Kullanıcının metin kutusu içersine ne yazdığını saptamak için metin özellikleri okunabilir veya düzenlenebilir.

Label (Etiket)

Daha çok kullanıcıya form üzerinde bilgi vermek için kullanılır. Bu nesneye kullanıcı tarafından giriş yapılamaz. Kullanıcının giriş yapabilmesi haricinde text kutusuna çok benzer.

Command Button (Komut Düğmesi)

Command Butonu bilgisayar programıyla kullanıcı arasındaki etkileşim için en kolay yolu geliştirir. Command butonu Windows'ta hazırdır. Bunlar düzenli butonlardır. Tıklanabilir veya klavye ile istenen şey yapılabilir.

CheckBox (İşaret Kutusu)

CheckBox kontrolü kullanıcıya iki seçenek sağlamak için kullanılır. Check kutuları kontrollü veya kontrolsüz olabilir. Denetimin değer özelliği kontrol edilerek herhangi bir zamanda Check kutusunun durumunun ne olduğu anlatılabilir.

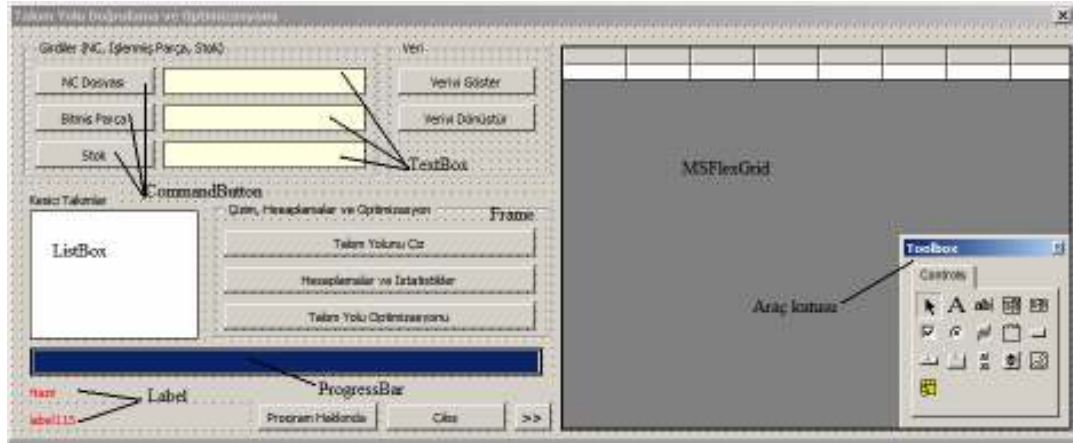
ListBox (Listeleme Kutusu)

ListBox kontrolü ComboBox kontrolüne çok benzer. Başlıca fark listenin her zaman görünür olmasıdır. Liste kutuları değerlerin hazır listeden kullanıcı tarafından istendiği zaman hazır olarak seçebilmesi için kullanılır.

ComboBox (Açılan Liste)

ComboBox kontrolü kullanıcıya belirli bir boşluk içinde bulunan nesnelerin hazır listeden seçilmesine olanak sağlamak isteniyorsa kullanılabilir. Genellikle tek objeyi gösterir, ama sahip olduğu “drop-down” oku sayesinde seçeneklerin bir listesi görülebilir.

Kontrol çeşitlerinin bulunduğu bir uygulama Şekil 5.15’ verilmiştir.



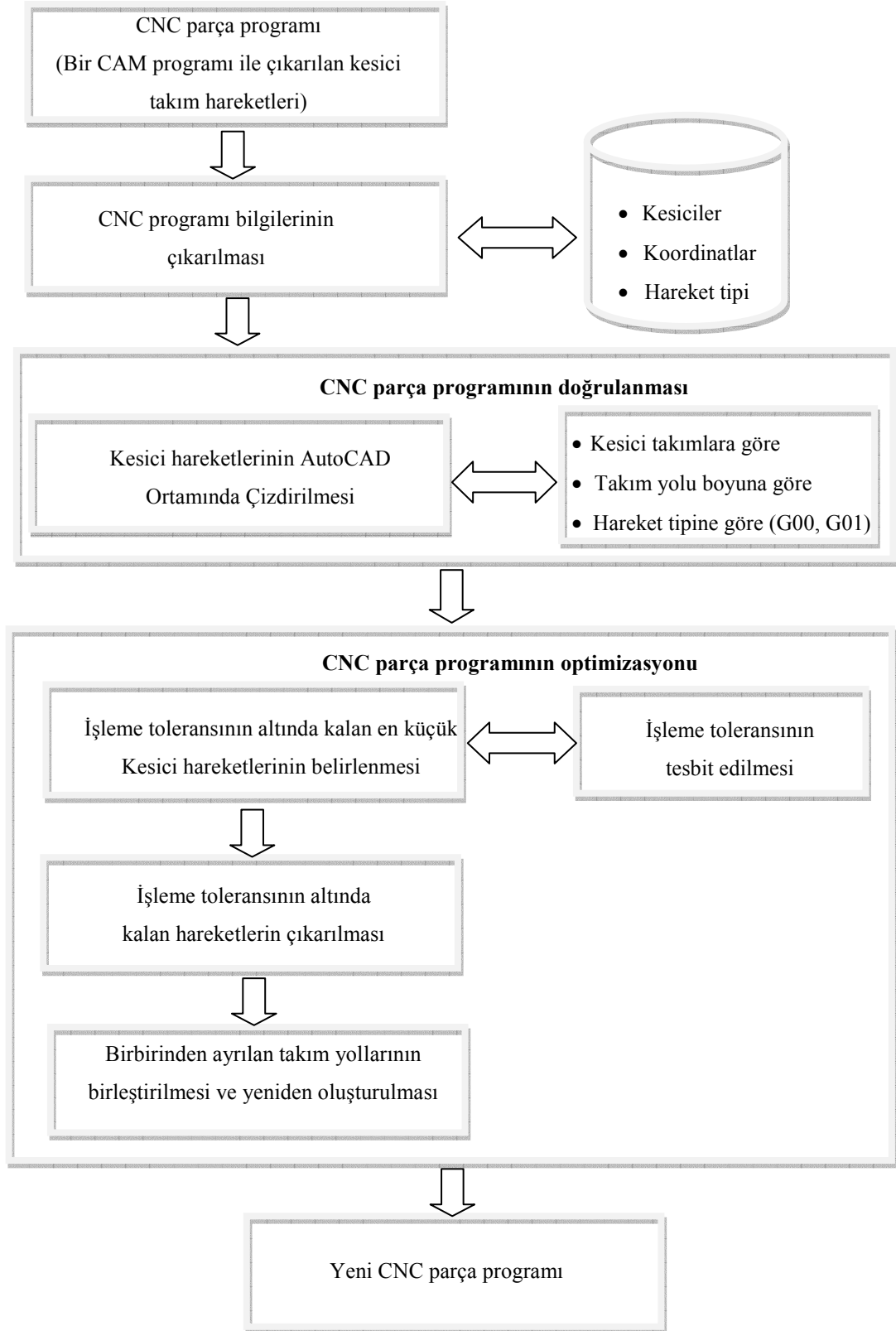
Şekil 5.15. VBA kontrol çeşitleri

6. CNC PARÇA PROGRAMLARI İÇİN TAKIM YOLU DOĞRULAMASI VE OPTİMİZASYONU

Üretimi yapılacak parçanın CNC tezgahlarında işlenmesi için CAM programlarında, CNC kodları çıkarılmaktadır. Bu kodlar tezgaha aktarılarak işleme gerçekleştirilmektedir. CAM programlarında CNC kodları çıkarılırken; dalma derinliği, yanal kayma, işleme toleransı gibi parametreler kullanılmaktadır. CNC datası bu parametrelere göre hazırlanır. Bu çalışmada, CAM parametrelerinden olan, işleme toleransının takım yolu üzerindeki etkileri araştırılmaktadır. İşleme toleransı artırılıp azaldıkça takım yolu boyundaki değişimler incelenmekte ve işleme tolerans değerine göre oluşan takım yolu boyları analiz edilebilmektedir. Serbest sitil yüzeylerin işlenmesi düz yüzeylerin işlenmesine göre daha fazla karmaşıklık gösterebilmektedir. Böyle durumlarda CNC verisinde işleme toleransından küçük takım yolları oluşmaktadır.

Tezgaha gönderilecek CNC kodlarının kontrol edilmesi daha doğru bir üretim imkanı sağlanmaktadır. Bu kontrol, varolan CNC verisinin AutoCAD ortamında çizdirilmesiyle gerçekleştirilir. AutoCAD Layer ve List özellikleri kullanılarak takım yolları detaylı olarak incelenebilmektedir. Takım yolu çizdirilmesiyle kullanıcıya toplam takım yolu boyu, en büyük takım yolu boyu, en küçük takım yolu boyu, toplam takım sayısı gibi bilgiler rapor edilmektedir. Bu bilgiler doğrultusunda işleme toleransından küçük takım yolları silinerek çıkarılabilmekte ve yeni takım yolları eklenerek optimize edilmiş CNC datası oluşturulmaktadır.

Yapılan bu çalışma iki aşama halinde ele alınmıştır. Birinci aşama sayısal denetim takım yolunun okunarak koordinat bilgilerinin çıkarılması ve bilgisayar ortamında çizdirilmesidir. İkinci aşama da mevcut takım yolunun işleme toleransına göre optimize edilmesidir. Şekil 6.1’de CNC parça programları için takım yolu doğrulanması ve optimizasyonuna yönelik yapılan çalışmanın aşamaları gösterilmektedir.



Şekil 6.1. CNC Parça programları için takım yolu doğrulanması ve optimizasyonu

Geliştirilen programın komut satırları tezin ekler kısmında verilmiştir

6.1. CNC parça program bilgilerinin çıkarılması

Sayısal denetim verisinin her satırında araştırma yapılarak bütün fonksiyonlar ve değerler bulunmaktadır. Böylelikle program G (Hareketli kodlar), M (Yardımcı kodlar), S (devir), F (İlerleme) fonksiyonlarını, takımları ve X, Y ve Z koordinatlarının değerlerini okumaktadır. Şekil 6.2’de varolan bir CNC parça programının kodları gösterilmektedir.

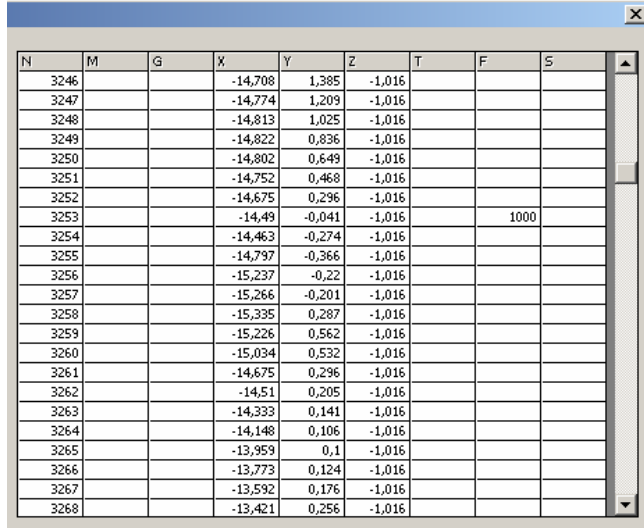
```

%
N1 G49 G54 G20 G80 G40 G90 G23 G94 G17 G98
N2 T1 M6
N3 G0 X-.308 Y-30.6 S70 M3
N4 G43 Z9.1 H1
N5 G1 Z-1. F300.
N6 Y-25.
N7 X-25. F1000.
N8 Y25.
N9 X25.
N10 Y-25.
N11 X-.308
N12 Y-23.75
N13 Y-22.5
N14 X-22.5
N15 Y22.5
N16 X22.5
N17 Y-22.5
N18 X-.308
N19 Y-21.25
N20 Y-20.
N21 X-20.
N22 Y20.
N23 X20.
N24 Y-20.
N25 X-.308
N26 Y-18.75
N27 Y-17.5
N28 X-17.5
N29 Y17.5
N30 X17.5
N31 Y-17.5
N32 X-.308
N33 Y-16.25
N34 Y-15.
N35 X-15.
N36 Y15.
N37 X15.
N38 Y-15.
N39 X-.308
N40 Y-14.221
N41 Y-13.441
N42 X-5.067
N43 X-5.583 Y-13.498
N44 X-6.311 Y-13.47
N45 X-8.134 Y-13.132
N46 X-8.972 Y-12.812

```

Şekil 6.2. Varolan CNC kodları

Bu kodlar arasından fonksiyon değerleri aranarak sayısal denetim verisine dönüştürülerek tüm bilgiler Şekil 6.3'deki gibi bir tablo haline getirilir. Program her sayısal denetim veri satırı için bu fonksiyonların yanındaki sayısal karakteri aramaktadır.



N	M	G	X	Y	Z	T	F	S
3246			-14,708	1,385	-1,016			
3247			-14,774	1,209	-1,016			
3248			-14,813	1,025	-1,016			
3249			-14,822	0,836	-1,016			
3250			-14,802	0,649	-1,016			
3251			-14,752	0,468	-1,016			
3252			-14,675	0,296	-1,016			
3253			-14,49	-0,041	-1,016		1000	
3254			-14,463	-0,274	-1,016			
3255			-14,797	-0,366	-1,016			
3256			-15,237	-0,22	-1,016			
3257			-15,266	-0,201	-1,016			
3258			-15,335	0,287	-1,016			
3259			-15,226	0,562	-1,016			
3260			-15,034	0,532	-1,016			
3261			-14,675	0,296	-1,016			
3262			-14,51	0,205	-1,016			
3263			-14,333	0,141	-1,016			
3264			-14,148	0,106	-1,016			
3265			-13,959	0,1	-1,016			
3266			-13,773	0,124	-1,016			
3267			-13,592	0,176	-1,016			
3268			-13,421	0,256	-1,016			

Şekil 6.3. Tablo haline getirilen sayısal denetim takım yolları

6.2. CNC Parça Programının Doğrulanması

CAM programlarında hazırlanmış CNC verisi alınarak AutoCAD ortamında çizdirilmesiyle sayısal denetim verisinin doğrulanması yapılabilmektedir. AutoCAD ortamında çizim; CNC verisinde kullanılan takımlara göre, takım yolu boyuna göre ve takım yolu hareketine göre üç başlık altında yapılabilmektedir. Seçilen duruma göre çizim yapılırken AutoCAD Layer özelliği kullanılarak takım yolları gruplandırılır. Çizim sonunda bu gruplardan istenilen aktifleştirilip istenilen saklanarak takım yollarını detaylı olarak inceleyebilmektedir. Ayrıca AutoCAD List özelliği kullanılarak da her bir takım yolu seçilebilmekte ve bu takım yolu koordinatları ve boyu gibi özellikler öğrenilebilmektedir.

6.2.1. Sayısal denetim takım yolunun AutoCAD yazılımında çizdirilmesi

Şekil 6.5'deki gibi tablo haline dönüştürülen takım yolları çizime hazırdır. AutoCAD ortamında takım yollarının çizdirilmesi sırası ile aşağıda anlatılmıştır.

1. Takımlara göre çizim yapılması

CNC verisinde kullanılmış takımların işlemelerine göre takım yollarını gruplayarak AutoCAD Layer özelliğine atar. Her takım için, takım yolu farklı renklerde çizdirilir. Kullanıcı istediği katmanı aktifleştirerek takım yolunu detaylı olarak incelenebilir.

2. Takım yolu boyuna göre çizim yapılması

İkinci takım yolu çizim metodu takım yolu boyuna göre çizim yapılmasıdır. CNC verisindeki takım yolları, alt sınır işleme toleransı olarak üst sınır da tayin edilen aralıkta belirlenerek 10 guruba ayrılır. Böylelikle her bir grup bir katmana atanarak takım yolu çizdirilir. işleme toleransının altındaki takım yolları da farklı bir katmana atandığından bu takım yolları seçilerek kaç tane oldukları List özelliği ile hesaplanır.

3. Takım hareketlerine göre çizim yapılması

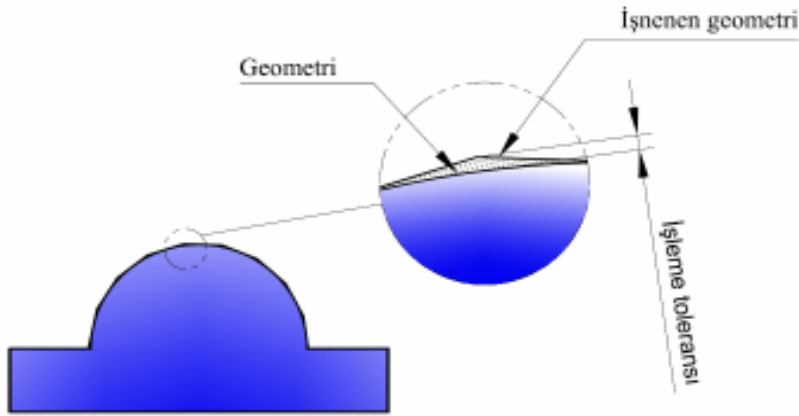
Bir diğer çizim metodu da takım hareketlerine göre gruplamanın yapılmasıdır. Bu durumda G00 ve G01 hareketleri farklı katmanlara atanarak çizim yaptırılır. İstenilen katman aktifleştirilerek takım hareketleri detaylı olarak incelenebilmektedir.

6.3. CNC Parça Programının Optimizasyonu

Bu çalışmada, CAM programlarında takım yolu oluşturulurken kullanılan parametrelerden biri olan işleme toleransına göre optimizasyon yapılmaktadır.

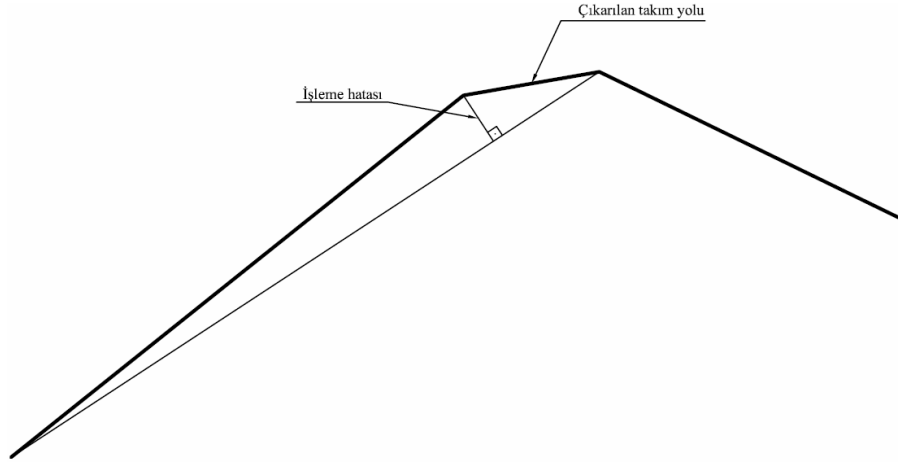
İşleme toleransı

İşleme toleransı orjinal parça ile işleme sonrasında oluşan parça arasındaki farktır. Bu değer bitirme operasyonlarında küçük tutulur. İşleme toleransı küçüldükçe takım yolu sayısı artar. İşleme toleransı yüzey kalitesini etkiler. Şekil 6.4’de işleme toleransı gösterilmektedir.



Şekil 6.4. İşleme toleransı

Kaba işlemlerde asıl amaç, az zamanda fazla talaş kaldırarak işleme gerçekleştirebilmek olduğundan işleme toleransı, kaba işlemlerde bitirme işlemeye göre daha büyük tutulur. Yapılan incelemeler göstermiştir ki, BDÜ yazılımları, girilen işleme toleransına göre, takım yolu boyutları uzamakta yada kısalmakta ve işleme toleransından daha küçük boyutlarda takım yolu çıkarmaktadır. CNC verisinin incelenmesiyle, CNC verisinde var olan işleme toleransından küçük takım yolları bulunup çıkarılarak yeni takım yolları eklenmekte ve bu yeni takım yollarına göre yeni CNC verisi oluşturulmaktadır. İşleme toleransından küçük takım yolu çıkarılacakken oluşabilecek işleme hataları da hesaplanmaktadır. Şekil 6.5’de iki takım yolundan küçük olanın çıkarılması sonucu oluşacak yüzey hatası görülmektedir.



Şekil 6.5. Takım yolunun çıkarılması ile oluşan hata

İşleme toleransından küçük takım yolunun çıkarılmasıyla oluşan hata oranı işleme toleransını geçmemelidir. En küçük hata ve en büyük hatanın ne kadar olduğu programda hesaplanmaktadır.

En küçük hata

En küçük hatanın sıfır çıkması durumudur. Aynı doğrultuda ardı ardına bulunan ve işleme toleransından küçük iki takım yolunun birleştirilmesi halinde oluşmaktadır.

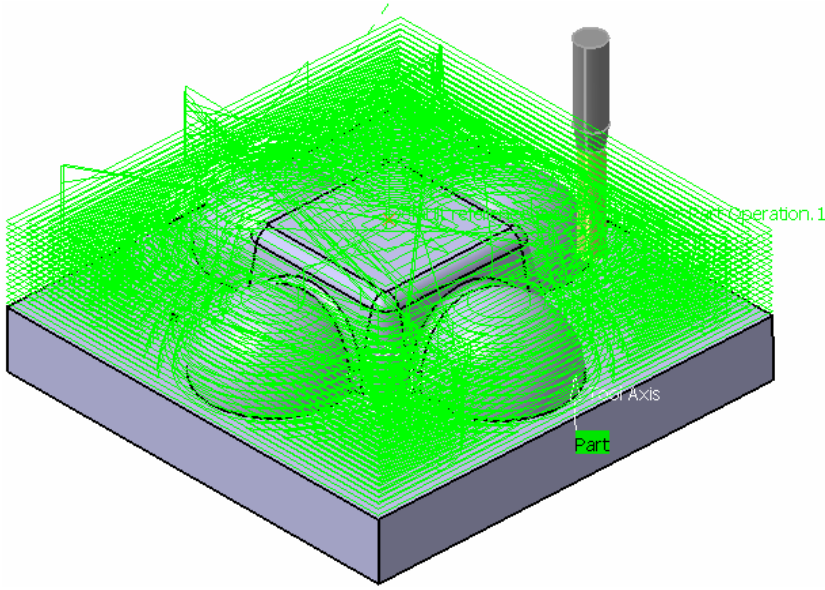
Aşağıda 0.1 mm işleme toleransı için yapılan örnek bir uygulama verilmektedir. Takım yolu için girilen işleme parametreleri şu şekildedir.

İşleme toleransı = 0,1 mm,

Yanal kayma mesafesi = $0,5 \times \text{takım çapı} = 0,5 \times 10 = 5 \text{ mm}$,

Dalma derinliği = 3 mm,

Bu parametreler girilerek elde edilen takım yolu Şekil 6. 6'de görülmektedir.



Şekil 6.6. İşleme toleransı = 0,1 mm, Yanal kayma mesafesi = $0,5 \times \text{takım çapı} = 0,5 \times 10 = 5$ mm, Dalma derinliği = 3 mm olarak oluşturulmuş takım yolu

Üretilen takım yolu, takım yolu boylarına göre incelendiğinde 6. 1’deki değerler elde edilecektir.

Çizelge 6.1. Takım yolunun boylarına göre gruplandırılması

Takım Yolu Boyu	Takım Yolu Adedi
$a > 5$	3664
$5 > a > 4,3$	436
$4,3 > a > 3,6$	452
$3,6 > a > 2,9$	1427
$2,9 > a > 2,2$	696
$2,2 > a > 1,5$	802
$1,5 > a > 0,8$	2739
$0,8 > a > 0,1$	3451
$a < 0,1$	78

Çizelge 6.1’de görüldüğü gibi girilen 0,1 mm takım yolundan küçük takım yolu sayısı 78 tanedir. 0,1 mm’den küçük takım yollarını çıkarmak için çıkarılan takım

yolunun ne kadarlık işleme hatasına sebep olacağının bilinmesi gerekir. Yukarıdaki işleme örneğinde işleme hatalarının bir kısmı Çizelge 6.2’de takım boyları ile beraber verilmiştir.

Çizelge 6.2. Takım yolu boyu

Boy	Hata		Boy	Hata		Boy	Hata		Boy	Hata
0,0765	0,0037		0,0925	0,0000		0,0588	0,0000		0,0149	0,0000
0,0289	0,0024		0,0196	0,0000		0,0117	0,0117		0,0890	0,0059
0,0587	0,0474		0,0360	0,0224		0,0990	0,0813		0,0504	0,0370
0,0397	0,0000		0,0801	0,0000		0,0466	0,0306		0,0570	0,0156
0,0295	0,0289		0,0166	0,0018		0,0165	0,0165		0,0709	0,0614
0,0625	0,0343		0,0687	0,0427		0,0698	0,0000		0,0631	0,0316
0,0615	0,0602		0,0569	0,0000		0,0949	0,0142		0,0173	0,0086

Bu uygulamada hata ile bazı istatistiksel değerler şu şekildedir.

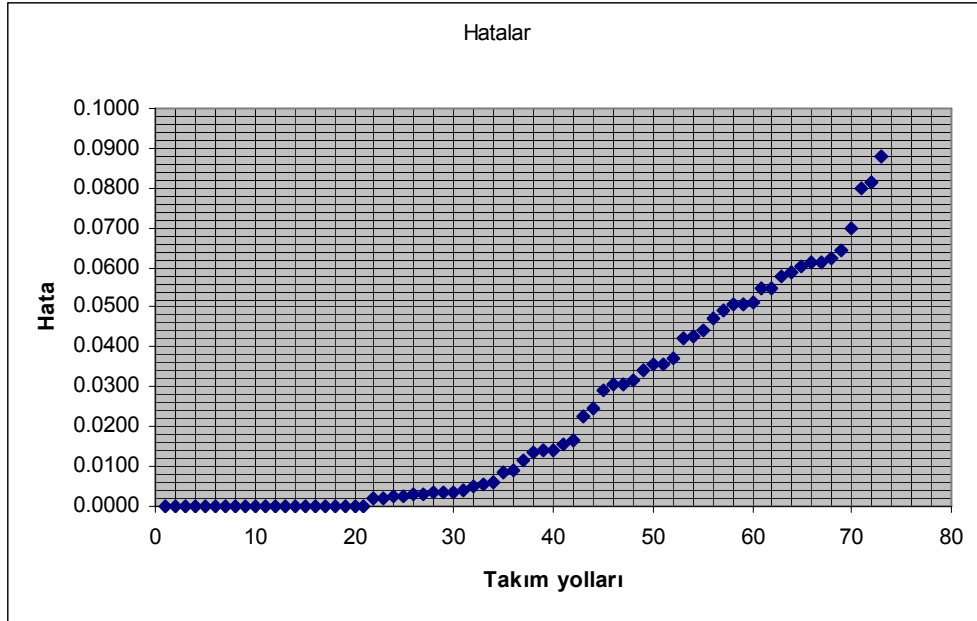
Aritmetik ortalama = 0,0232 mm,

Standart sapma = 0,0257 mm,

En küçük hata = 0 mm (21 adet),

En büyük hata = 0,0878

Grafiksel olarak hatalar verildiğinde Şekil 6. 7’deki gibi olacaktır.



Şekil 6.7. Takım yolu hataları

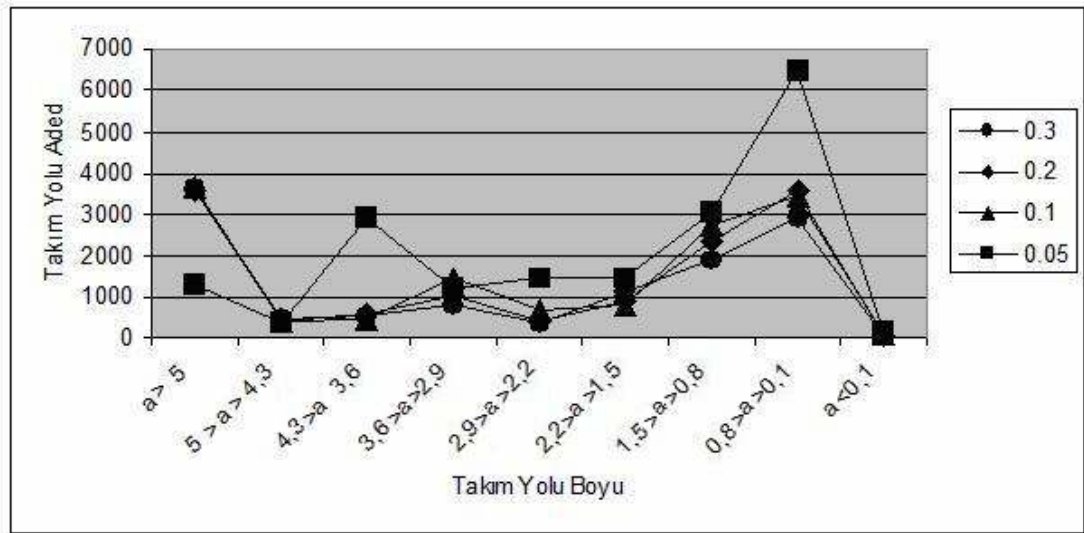
Bu grafikte de görüldüğü gibi değerlerin büyük çoğunluğu 0 – 0,0600 mm arasında toplanmıştır. Kaba işleme operasyonu olduğu için bu hatalar yok sayılıp takım yolları sayısal denetim kodundan çıkarılabilir. Çıkarma işlemi yapılırken hareket tipine de dikkat edilmelidir. G00 veya G01 hareketlerinin başlangıç satırı çıkarılmamalıdır. Yazılan koda bu ayırım eklenmiştir.

Daha geniş kapsamlı bir örnek vererek BDÜ yazılımlarında 0.3, 0.2, 0.1, 0.05 işleme toleransına bağlı olarak oluşturulan minimum takım yolu boyutları çizelge 6.3’de verilmektedir. Bu çizelgede her bir işleme toleransına göre hazırlanmış takım yollarının alt sınır 0.1 mm ve üst sınır da 5 mm olarak girilen takım yolu boyutlarının gruplandırılmasıyla oluşan takım yolu sayıları gösterilmektedir.

Çizelge 6.3. Farklı işleme toleranslarında, takım yollarının takım yolu boyuna göre gruplandırılması

Takım Yolu Boyu	İşleme Toleransı / Takım Yolu Adedi			
	0.3	0.2	0.1	0.05
$a > 5$	3606	3595	3664	1293
$5 > a > 4,3$	473	424	436	376
$4,3 > a > 3,6$	493	561	452	2976
$3,6 > a > 2,9$	793	1043	1427	1212
$2,9 > a > 2,2$	360	402	696	1461
$2,2 > a > 1,5$	1088	873	802	1428
$1,5 > a > 0,8$	1920	2408	2739	3052
$0,8 > a > 0,1$	2967	3566	3451	6494
$a < 0,1$	53	75	78	136

Çizelgede verilen değerler grafik halinde Şekil 6.8’de gösterilmektedir.



Şekil 6.8. Farklı işleme toleranslarında takım yolu boyuna göre takım yolunun grafik olarak gösterilmesi

Şekilde de görüldüğü gibi işleme toleransı 0.3’e kadar arttırıldığı durumlarda da 0.1’den daha küçük takım yolları türetilmiştir. Bununla birlikte her işleme toleransının kendi değerinden küçük takım yolları Çizelge 6.4 ‘de verilmektedir.

Çizelge 6.4. Farklı işleme toleranslarında, işleme toleransından küçük takım yolları

Takım Yolu Boyu	İşleme Toleransı / Takım Yolu Adedi			
	0.3	0.2	0.1	0.05
$a < 0.3$	671			
$a < 0.2$		189		
$a < 0.1$			78	
$a < 0.05$				61

Çizelgede de görüldüğü gibi takım yolunda işleme toleransından küçük takım yolları oluşmaktadır. Bu takım yollarının programda bulunup çıkarılmasıyla, işleme toleransına bağlı olarak takım yolu optimizasyonu yapılacak ve yeni takım yolu oluşturulacaktır.

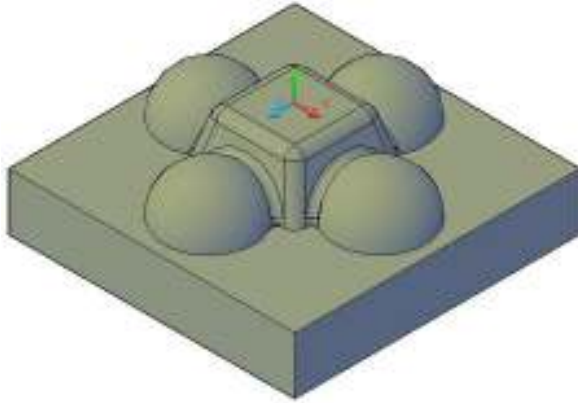
Yapılan optimizasyon sonrasında, silinen satırlar, baştan sona kadar tektek gözden geçirilir. Program satır numaraları tekrardan düzenlenir. Böylelikle yeni bir CNC programı elde edilir. Bu yeni CNC programı satır sayısı olarak azalmaktadır.

7. CNC PARÇA PROGRAMLARI İÇİN TAKIM YOLU DOĞRULAMASI VE OPTİMİZASYON YÖNTEMİNDE ÖRNEK UYGULAMALAR

CNC parça programları için takım yolu doğrulanması ve optimizasyonu için yapılan bu çalışmada, farklı parça modellerinin farklı işleme toleranslarında CNC kodları çıkarılmıştır. Bu kodların programda doğrulanması ve çıkan sonuca göre optimizasyonunun yapılmasıyla elde edilen örnek uygulamalar aşağıda sırasıyla verilmektedir. CNC kodlarındaki işleme tolerans değerleri; 0.3 mm, 0.2 mm, 0.1 mm, 0.05 mm olarak tanımlanmıştır. İşleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya; okunan satır sayısı (adet), toplam takım yolu boyu (mm), en küçük takım yolu boyu (mm), en büyük takım yolu boyu (mm) ve toplam takım sayısı (adet) rapor edilmektedir.

7.1. Örnek Uygulama 1

Şekil 7.1’de takım yolu doğrulanması ve optimizasyonu yapılan ilk parça modeli verilmektedir.



Şekil 7.1. Takım yolu doğrulanması ve optimizasyonu yapılan parça modeli

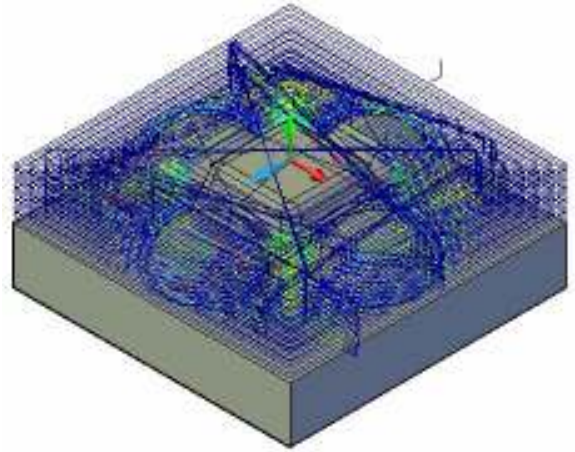
İşleme toleransı 0.3 mm olan takım yolu

0.3 mm işleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya sunulan takım yolu sonuçları Şekil 7.2’de verilmektedir.



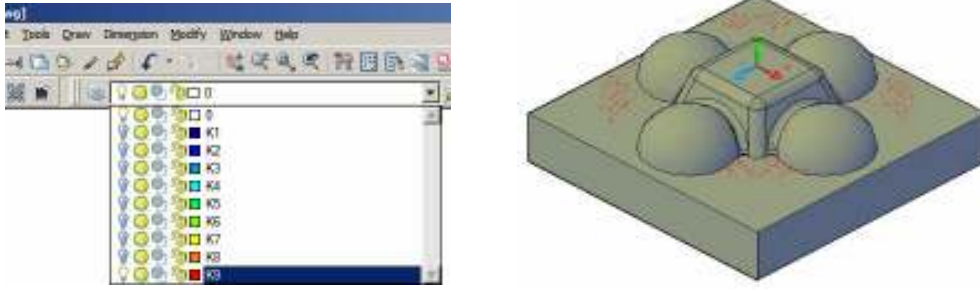
Şekil 7.2. İşleme toleransı 0.3 mm olan takım yolu sonuçları

0.3 işleme toleransına göre oluşturulmuş takım yolunun, AutoCAD ortamında takım yolu boylarına göre gruplandırılarak çizilmesi Şekil 7.3’de verilmektedir.



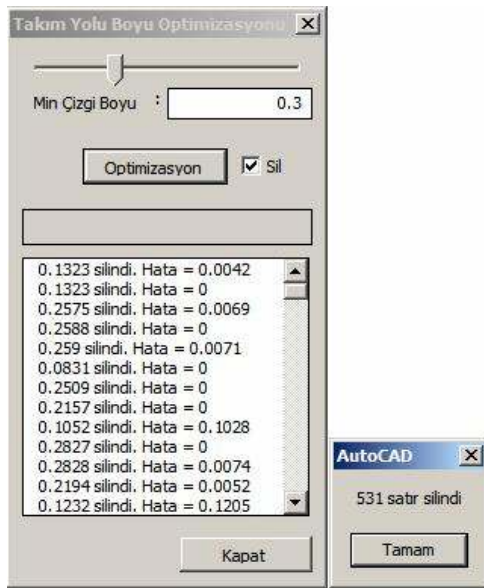
Şekil 7.3. İşleme toleransı 0.3 mm olan takım yolu

Takım yollarının çizdirilmesiyle her bir aralık bir katmana atanmaktadır. İncelenmek istenen takım yolu aralığı, bulunduğu katman aktifleştirilerek gerçekleştirilmektedir. Şekil 7.4’de işleme toleransı 0.3 mm’den küçük olan takım yollarının bulunduğu katman seçimi ve bu takım yollarının gösterilmesi verilmektedir.



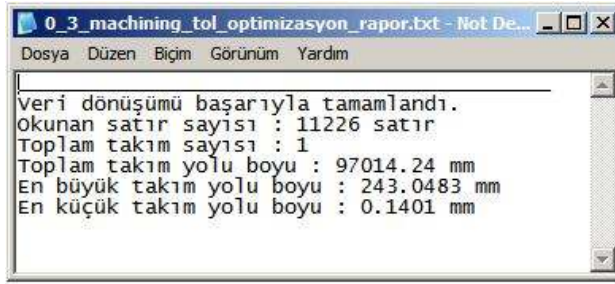
Şekil 7.4. İşleme toleransı 0.3 mm'den küçük takım yollarının bulunduğu katmanlar ve takım yollarının gösterilmesi

Yüzey hataları hesaplanarak yapılan takım yolu optimizasyonu ile 0.3 mm' den küçük takım yollarının bulunması Şekil 7.5'de gösterilmektedir.



Şekil 7.5. İşleme toleransı 0.3 mm'den küçük takım yollarının bulunması

İşleme toleransından küçük takım yollarının optimize edilerek çıkarılmasıyla oluşan yeni takım yolunun tekrar çizilmesiyle kullanıcıya verilen optimizasyon sonucu Şekil 7.6'da gösterilmektedir.



Şekil 7.6. İşleme toleransı 0.3 mm'ye göre optimizasyon yapılmış takım yolu sonuçları

Böylelikle işlem toleransı 0.3 mm olarak hazırlanmış CNC kodlarının doğrulanması ve çıkan sonuçların değerlendirilmesiyle yapılan optimizasyon tamamlanmıştır. Varolan CNC kodlarıyla optimizasyon sonucunda oluşan CNC kodlarının karşılaştırılması Çizelge 7. 1 'de gösterilmektedir.

Çizelge 7.1. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması

Karşılaştırılan Değerler	Varolan CNC Datası	Optimizasyon
Okunan satır sayısı (satır)	11757	11226
Toplan takım yolu boyu (mm)	97026.56	97014.24
En büyük takım yolu boyu (mm)	243.0483	243.0483
En küçük takım yolu boyu (mm)	0.0046	0.1401
Toplam takım sayısı (adet)	1	1

Bu tablodan da anlaşıldığı gibi optimizasyon sonucu CNC kodlarındaki satır sayısında azalma olmuştur. Bununla birlikte toplam takım yolu boyu ve en küçük takım yolu boyu da azalmıştır.

Var olan takım yolunun CNC kodları Şekil 7.7.a'da ve optimizasyon sonucu oluşturulmuş takım yolunun CNC kodları Şekil 7.7.b'de gösterilmektedir.

%
N1 G49
N2 T3 M6
N3 G0 X121.3 Y.7483 S5000 M3
N4 G43 Z7.08 H3
N5 G1 Z-2.92 F300.
N6 X110.
N7 Y-110. F2500.
N8 X-110.
N9 Y110.
N10 X110.
N11 Y.7483
N12 X107.
N13 X104.
N14 Y-104.
N15 X-104.
N16 Y104.
N17 X104.
N18 Y.7483
N19 X101.
N20 X98.

N11738 X3.4237 Y-105.7815
N11739 X-2.7164 Y-105.8387
N11740 X-10.0981 Y-104.649
N11741 X-16.3564 Y-102.4744
N11742 X-21.8854 Y-99.4869
N11743 X-27.8979 Y-94.8892
N11744 X-32.671 Y-89.5983
N11745 X-36.3361 Y-83.7432
N11746 X-39.0067 Y-77.2493
N11747 X-40.5696 Y-70.1911
N11748 X-41.6229 Y-68.6862 Z-40.3749
N11749 X-43.4318 Y-68.3667 Z-39.8699
N11750 Z-29.8699 F1000.
N11751 G0 Z.6
N11752 M5
N11753 M30
N11754 M2
N11757 M30
%

a.

%
N0 G49
N1 M6 T3
N2 M3 G0 X121.3 Y0.7483 S5000
N3 G43 Z7.08
N4 G1 Z-2.92 F300
N5 X110
N6 Y-110 F2500
N7 X-110
N8 Y110
N9 X110
N10 Y0.7483
N11 X107
N12 X104
N13 Y-104
N14 X-104
N15 Y104
N16 X104

N11210 X-21.8854 Y-99.4869
N11211 X-27.8979 Y-94.8892
N11212 X-32.671 Y-89.5983
N11213 X-36.3361 Y-83.7432
N11214 X-39.0067 Y-77.2493
N11215 X-40.5696 Y-70.1911
N11216 X-41.6229 Y-68.6862 Z-40.3749
N11217 X-43.4318 Y-68.3667 Z-39.8699
N11218 Z-29.8699 F1000
N11219 G0 Z0.6
N11220 M5
N11221 M30
N11222 M2
N11226 M30
%

b.

Şekil 7.7. Var olan CNC kodları ve optimizasyon sonucu oluşturulan CNC kodları karşılaştırılması a) Var olan CNC kodlar, b) Optimizasyon sonrası oluşturulan CNC kodları

Şekil 7.7’den de anlaşıldığı gibi varolan CNC kodları satır sayısı 11757 iken optimizasyon sonucu oluşturulmuş CNC kodları satır sayısı 11226’dır. Optimizasyon sonucu 531 satır silinerek yeni takım yolu oluşturulmuştur.

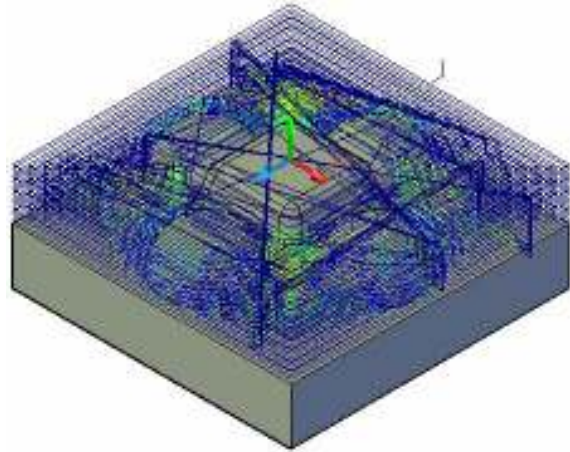
İşleme toleransı 0.2 mm olan takım yolu

0.2 mm işleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya sunulan rapor Şekil 7.8’de verilmektedir.



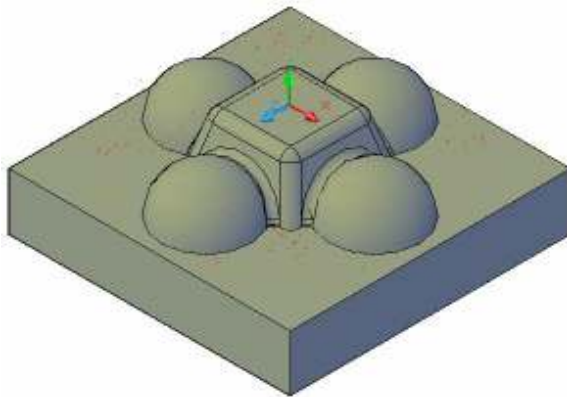
Şekil 7.8. İşleme toleransı 0.2 mm olan takım yolu sonuçları

0.2 işleme toleransına göre oluşturulmuş takım yolunun, AutoCAD ortamında takım yolu boylarına göre gruplandırılarak çizilmesi Şekil 7. 9’da verilmektedir.



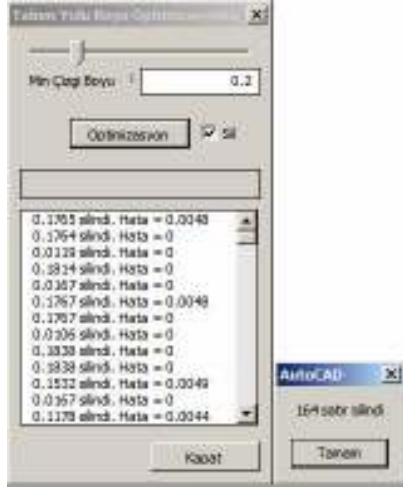
Şekil 7.9. İşleme toleransı 0.2 mm olan takım yolu

İşleme toleransı 0.2 mm’den küçük olan takım yollarının bulunduğu katman seçilmesiyle bu takım yollarının gösterilmesi Şekil 7.10’da verilmektedir.



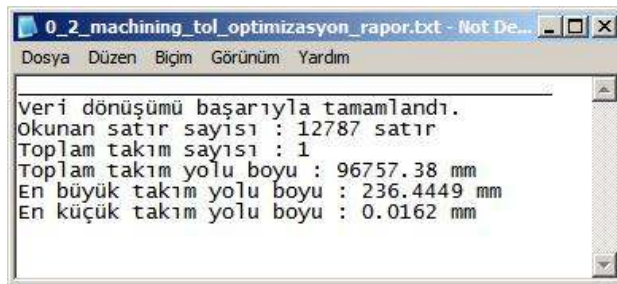
Şekil 7.10. İşleme toleransı 0.2 mm’den küçük takım yollarının gösterilmesi

Yüzey hataları hesaplanarak yapılan takım yolu optimizasyonu ile 0.2 mm' den küçük takım yollarının bulunması Şekil 7.11'de gösterilmektedir.



Şekil 7.11. İşleme toleransı 0.2 mm'den küçük takım yollarının bulunması

İşleme toleransından küçük takım yollarının optimize edilerek çıkarılmasıyla oluşan yeni takım yolunun tekrar çizilmesiyle kullanıcıya verilen optimizasyon sonucu Şekil 7.12'de gösterilmektedir.



Şekil 7.12. İşleme toleransı 0.2 mm'ye göre optimizasyon yapılmış takım yolu sonuçları.

Varolan CNC kodlarıyla optimizasyon sonucunda oluşan CNC kodlarının karşılaştırılması Çizelge 7. 2 'de gösterilmektedir.

Çizelge 7. 2. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması

Karşılaştırılan Değerler	Varolan CNC	Optimizasyon
Okunan satır sayısı (satır)	12951	12787
Toplam takım yolu boyu (mm)	96763.28	96757.38
En büyük takım yolu boyu (mm)	236.4449	236.4449
En küçük takım yolu boyu (mm)	0.0073	0.0162
Toplam takım sayısı (adet)	1	1

Bu tablodan da anlaşıldığı gibi optimizasyon sonucu CNC kodlarındaki satır sayısında azalma olmuştur. Bununla birlikte toplam takım yolu boyu ve en küçük takım yolu boyu da azalmıştır.

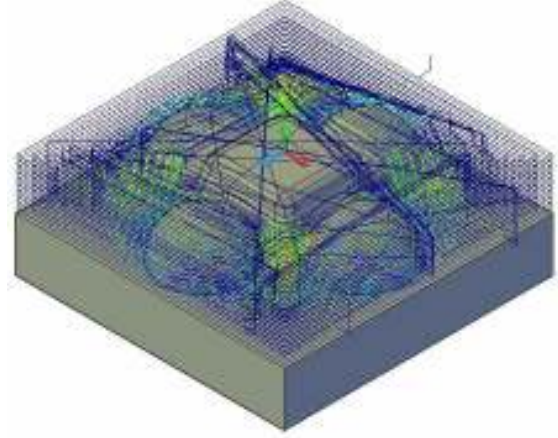
İşleme toleransı 0.1 mm olan takım yolu

0.1 mm işleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya sunulan rapor Şekil 7.13’de verilmektedir.



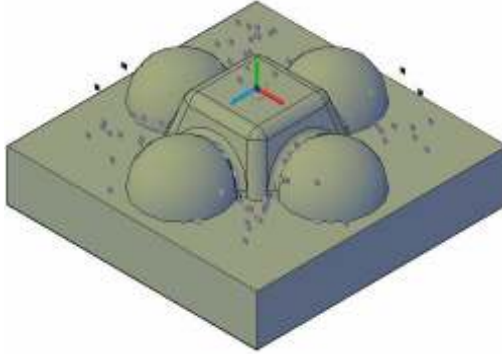
Şekil 7.13. İşleme toleransı 0.1 mm olan takım yolu sonuçları

0.1 işleme toleransına göre oluşturulmuş takım yolunun, AutoCAD ortamında takım yolu boylarına göre gruplandırılarak çizilmesi Şekil 7.14’de verilmektedir.



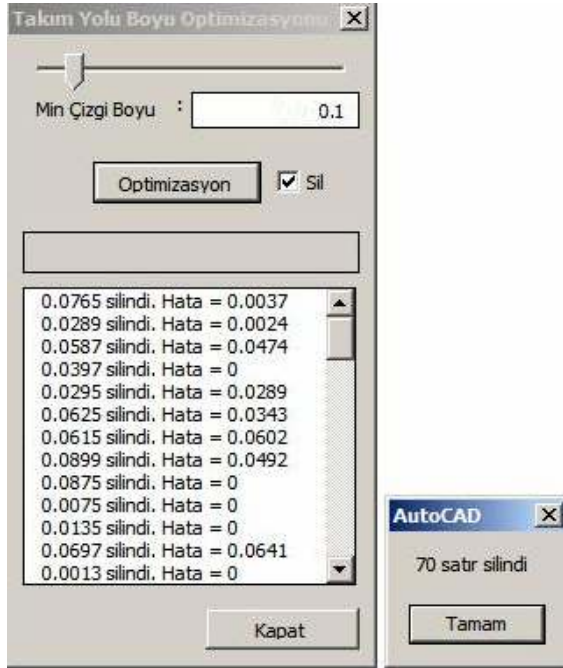
Şekil 7.14. İşleme toleransı 0.1 mm olan takım yolu

İşleme toleransı 0.1 mm'den küçük olan takım yollarının bulunduğu katman seçimi ve bu takım yollarının gösterilmesi Şekil 7.15'da verilmektedir.



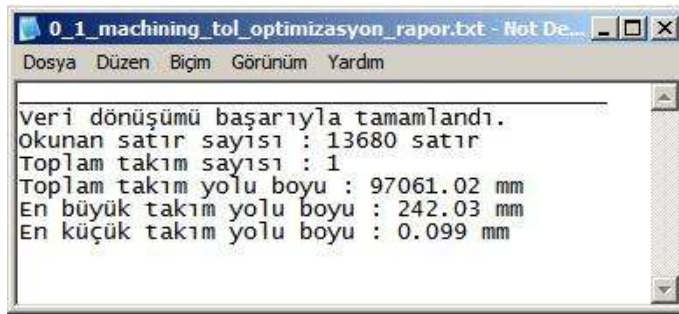
Şekil 7.15. İşleme toleransı 0.1 mm'den küçük takım yolları

Yüzey hataları hesaplanarak yapılan takım yolu optimizasyonu ile 0.1 mm'den küçük takım yollarının bulunması Şekil 7.16'da gösterilmektedir.



Şekil 7.16. İşleme toleransı 0.1 mm'den küçük takım yollarının bulunması

İşleme toleransından küçük takım yollarının optimize edilerek çıkarılmasıyla oluşan yeni takım yolunun tekrar çizilmesiyle kullanıcıya verilen optimizasyon sonucu Şekil 7.17'de gösterilmektedir.



Şekil 7.17. İşleme toleransı 0.1 mm'ye göre optimizasyon yapılmış takım yolu sonuçları

Varolan CNC kodlarıyla optimizasyon sonucunda oluşan CNC kodlarının karşılaştırılması Çizelge 7.3 'de gösterilmektedir.

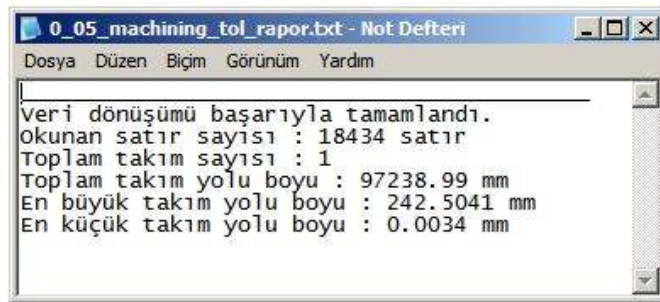
Çizelge 7.3. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması

Karşılaştırılan Değerler	Varolan CNC Datası	Optimizasyon
Okunan satır sayısı (satır)	13764	13680
Toplan takım yolu boyu (mm)	97062.33	97061.02
En büyük takım yolu boyu (mm)	242.03	242.03
En küçük takım yolu boyu (mm)	0.0013	0.099
Toplam takım sayısı (adet)	1	1

Bu tablodan da anlaşıldığı gibi optimizasyon sonucu CNC kodlarındaki satır sayısında azalma olmuştur. Bununla birlikte toplam takım yolu boyu ve en küçük takım yolu boyu da azalmıştır.

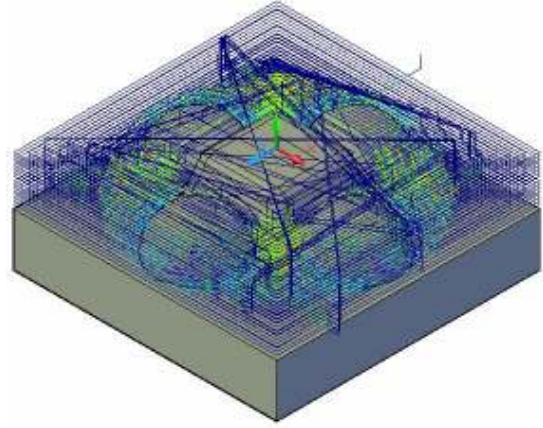
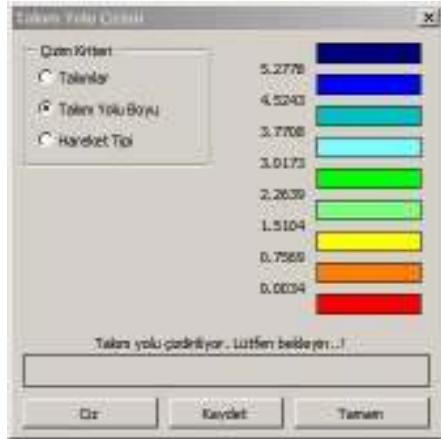
İşleme toleransı 0.05 mm olan takım yolu

0.05 mm işleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya sunulan rapor Şekil 7.18’de verilmektedir.



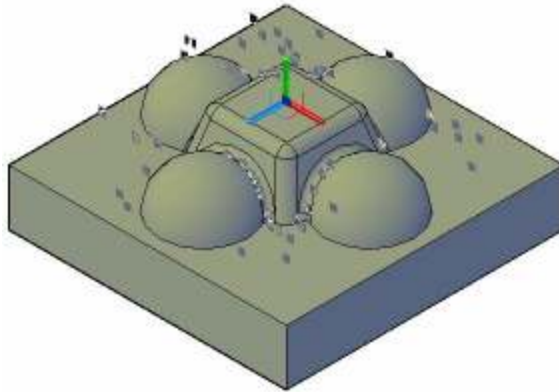
Şekil 7.18. İşleme toleransı 0.05 mm olan takım yolu sonuçları

0.05 işleme toleransına göre oluşturulmuş takım yolunun, AutoCAD ortamında takım yolu boylarına göre gruplandırılarak çizilmesi Şekil 7. 19’da verilmektedir.



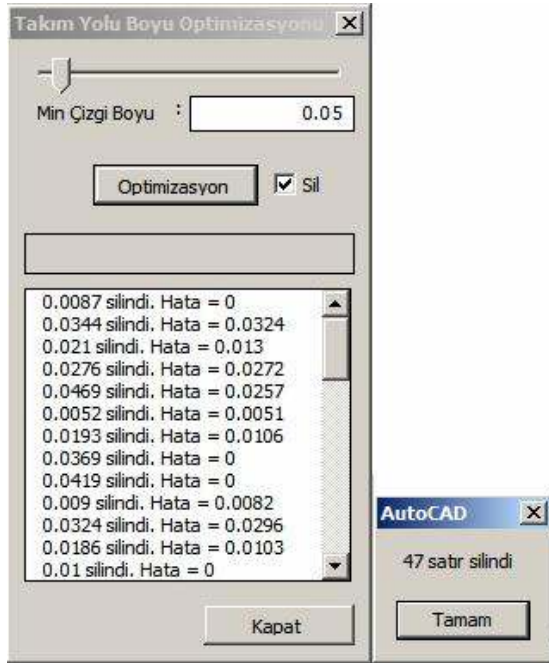
Şekil 7.19. İşleme toleransı 0.05 mm olan takım yolu

İşleme toleransı 0.05 mm'den küçük olan takım yollarının bulunduğu katman seçimi ve bu takım yollarının gösterilmesi Şekil 7.20'de verilmektedir.



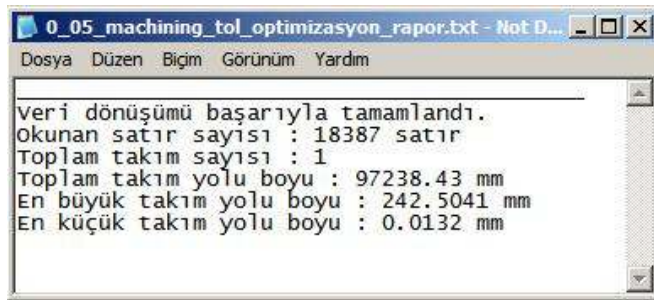
Şekil 7.20. İşleme toleransı 0.05 mm'den küçük takım yolları

Yüzey hataları hesaplanarak yapılan takım yolu optimizasyonu ile 0.05 mm' den küçük takım yollarının bulunması Şekil 7.21'de gösterilmektedir.



Şekil 7.21. İşleme toleransı 0.05 mm'den küçük takım yollarının bulunması

İşleme toleransından küçük takım yollarının optimize edilerek çıkarılmasıyla oluşan yeni takım yolunun tekrar çizilmesiyle kullanıcıya verilen optimizasyon sonucu Şekil 7.22'de gösterilmektedir.



Şekil 7.22. İşleme toleransı 0.05 mm'ye göre optimizasyon yapılmış takım yolu sonuçları

Varolan CNC kodlarıyla optimizasyon sonucunda oluşan CNC kodlarının karşılaştırılması Çizelge 7.4 'de gösterilmektedir.

Çizelge 7.4. Varolan CNC kodları ve optimizasyon sonucu CNC kodları değerlerinin karşılaştırılması

Karşılaştırılan Değerler	Varolan CNC Datası	Optimizasyon
Okunan satır sayısı (satır)	18434	18387
Toplam takım yolu boyu (mm)	97238.99	97238.43
En büyük takım yolu boyu (mm)	242.5041	242.5041
En küçük takım yolu boyu (mm)	0.0034	0.0132
Toplam takım sayısı (adet)	1	1

Bu tablodan da anlaşıldığı gibi optimizasyon sonucu CNC kodlarındaki satır sayısında azalma olmuştur. Bununla birlikte toplam takım yolu boyu ve en küçük takım yolu boyu da azalmıştır.

Aynı parçanın diğer parametreleri aynı olacak şekilde, işleme toleransları değiştirilerek oluşturulmuş bu takım yollarının karşılaştırılması Çizelge 7.5’de verilmektedir.

Çizelge 7.5. Farklı işleme toleranslarındaki varolan CNC kod sonuçlarının karşılaştırılması

İşleme toleransı (mm)	0.3	0.2	0.1	0.05
Okunan satır sayısı	11757	12951	13764	18434
Toplam takım yolu boyu	97026.56	96763.28	97062.33	97238.99
En büyük takım yolu	243.0483	236.4449	242.03	242.5041
En küçük takım yolu	0.0046	0.0073	0.0013	0.0034
Toplam takım sayısı	1	1	1	1

Bu çizelgedeki değerlerden anlaşıldığı gibi işleme toleransı küçüldükçe CNC kodlarındaki satır sayısı yani takım hareketleri artmaktadır. Çizelge 7.6’da aynı takım yollarının kendi işleme toleranslarına göre optimizasyonlarının karşılaştırılması sonuçları verilmektedir.

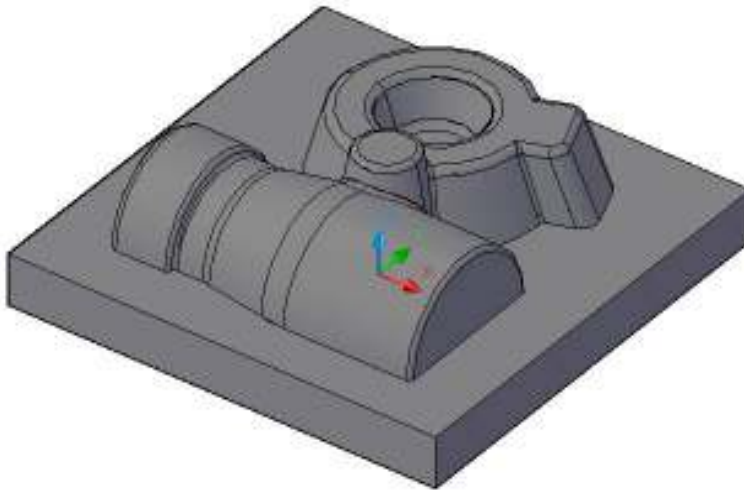
Çizelge 7.6. Farklı işleme toleranslarındaki takım yolları optimizasyon sonuçlarının karşılaştırılması.

İşleme toleransına göre	0.3	0.2	0.1	0.05
Okunan satır sayısı	11226	12787	13680	18387
Toplam takım yolu boyu	97014.24	96757.38	97061.02	97238.43
En büyük takım yolu	243.0483	236.4449	242.03	242.5041
En küçük takım yolu	0.1401	0.0162	0.099	0.0132
Toplam takım sayısı	1	1	1	1

Yukarıdaki iki çizelgede var olan CNC kodlarıyla, optimizasyon sonucunda oluşturulmuş CNC kodları karşılaştırılması verilmiştir. Optimizasyon sonucunda, satır sayısının azaldığı ve buna bağlı olarak da toplam takım yolu boyunun azaldığı tesbit edilmiştir. Toplam takım yolu boyu işleme zamanını doğrudan etkilediğinden toplam takım yolunun azalmasıyla işleme zamanının azaldığı öngörülmüştür.

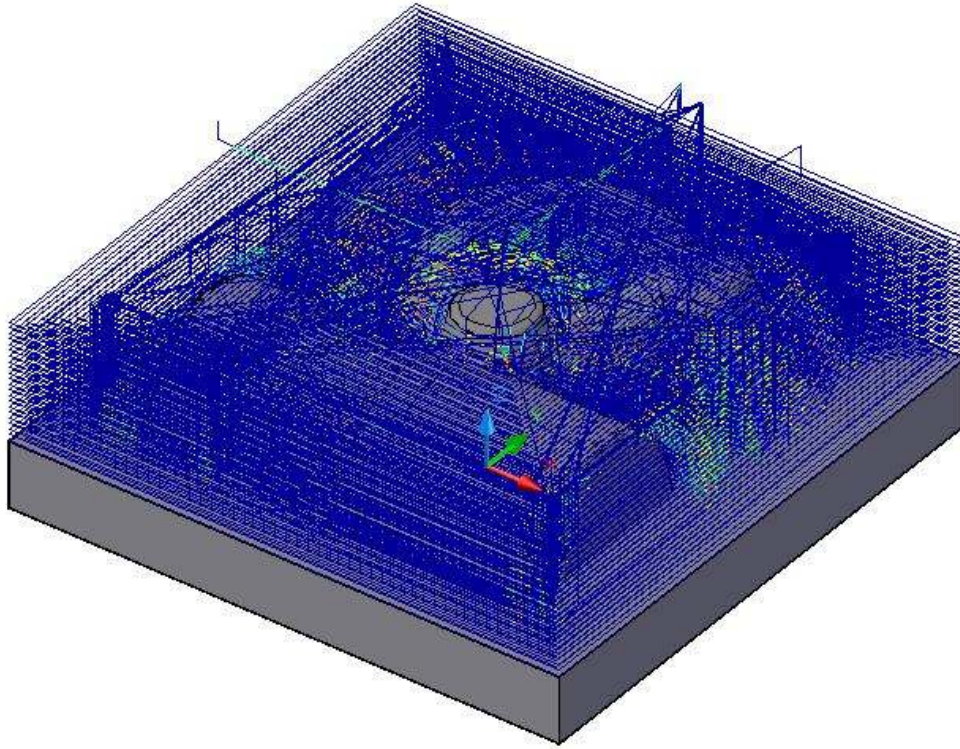
7.2 Örnek Uygulama 2

Şekil 7.23’de takım yolu doğrulaması ve optimizasyonu yapılan ikinci parça modeli verilmektedir.



Şekil 7.23. Takım yolu doğrulaması ve optimizasyonu yapılan parça modeli

CNC kodlarındaki işleme tolerans değerleri; 0.3 mm, 0.2 mm, 0.1 mm, 0.05 mm olarak tanımlanmıştır. İşleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya; okunan satır sayısı (adet), toplam takım yolu boyu (mm), en küçük takım yolu boyu (mm), en büyük takım yolu boyu (mm) ve toplam takım sayısı (adet) rapor edilmektedir. Buna göre 0.3 mm işleme toleransına göre çıkarılmış takım yolu şekil 7. 24’de gösterilmektedir.



Şekil 7.24. İşlem toleransı 0.3 mm olan takım yolu

Parçanın diğer parametreleri aynı kalmak kaydıyla, farklı işleme toleranslarında işlenmesiyle oluşan takım yolunda, takım yolu boyunun 0.1 mm ile 5 mm arasında gruplandırılarak gösterilmesi çizelge 7. 7’de verilmektedir.

Çizelge 7. 7. Farklı işleme toleranslarında takım yollarının, takım yolu boyuna göre gruplandırılması

Takım Yolu Boyu	İşleme Toleransı / Takım Yolu Adedi			
	0.3	0.2	0.1	0.05
$a > 5$	4126	4029	4057	2513
$5 > a > 4,3$	694	561	312	1661
$4,3 > a > 3,6$	600	777	910	798
$3,6 > a > 2,9$	671	847	985	536
$2,9 > a > 2,2$	1279	1339	1607	2431
$2,2 > a > 1,5$	1326	1344	1509	2552
$1,5 > a > ,8$	2475	2760	2984	3651
$0,8 > a > 0,1$	3426	4331	5893	9602
$a < 0,1$	83	75	149	240

Her bir işleme toleransına göre, işleme tolerans değerinden küçük takım yolu sayısı Çizelge 7.8’ de verilmektedir.

Çizelge 7.8. Farklı işleme toleranslarında takım yollarının, işleme toleranslarından küçük takım yolları

Takım Yolu Boyu	İşleme Toleransı / Takım Yolu Adedi			
	0.3	0.2	0.1	0.05
$a < 0.3$	743			
$a < 0.2$		357		
$a < 0.1$			149	
$a < 0.05$				78

Çizelgede de görüldüğü gibi işleme toleransından küçük takım yolları oluşmaktadır. İşleme toleransına göre ele alınan CNC parça programına ait bazı bilgiler Çizelge 7.9’da verilmiştir.

Çizelge 7.9. Farklı işleme toleranslarındaki varolan CNC kod sonuçlarının karşılaştırılması

İşleme toleransı (mm)	0.3	0.2	0.1	0.05
Okunan satır sayısı	14685	16068	18411	23989
Toplam takım yolu boyu	197136.94	196714.18	196674.32	197313.07
En büyük takım yolu	300	349.5068	336.3473	349.0141
En küçük takım yolu	0.0163	0.009	0.0098	0.002
Toplam takım sayısı	1	1	1	1

Bu çizelgedeki değerlerden anlaşıldığı gibi işleme toleransı küçüldükçe CNC programındaki satır sayısı, yani takım hareketleri artmaktadır. Çizelge 7.8’de aynı takım yollarının kendi işleme toleranslarına göre optimizasyonlarının karşılaştırılması sonuçları verilmektedir.

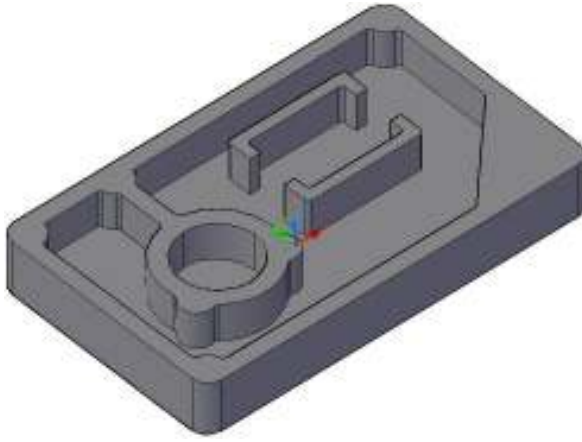
Çizelge 7.10. Farklı işleme toleranslarındaki takım yolları optimizasyon sonuçlarının karşılaştırılması

İşleme toleransına göre	0.3	0.2	0.1	0.05
Okunan satır sayısı	14056	15750	18326	23920
Toplam takım yolu boyu	197129.98	196709.81	196673.42	197312.57
En büyük takım yolu	300	349.5068	336.3473	349.0141
En küçük takım yolu	0.0656	0.0624	0.0282	0.0245
Toplam takım sayısı	1	1	1	1

Yukarıdaki iki çizelgede var olan CNC kodlarıyla, optimizasyon sonucunda oluşturulmuş CNC kodları karşılaştırılması verilmiştir. Optimizasyon sonucunda, satır sayısının azaldığı ve buna bağlı olarak da toplam takım yolu boyunun azaldığı tesbit edilmiştir. Toplam takım yolu boyu işleme zamanını doğrudan etkilediğinden toplam takım yolunun azalmasıyla işleme zamanının azaldığı öngörülmüştür.

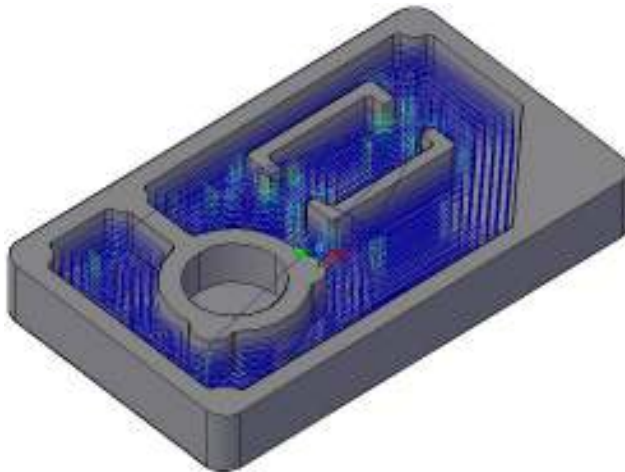
7.3. Örnek Uygulama 3

Şekil 7.25’de takım yolu doğrulaması ve optimizasyonu yapılan üçüncü parça modeli verilmektedir. Bu örnek uygulama da serbest sitil yüzeylerden ziyade pirizmatik yüzeylere sahip parça modeli seçilerek takım yolu doğrulaması ve optimizasyonu yapılmıştır.



Şekil 7.25. Takım yolu doğrulaması ve optimizasyonu yapılan parça modeli

CNC kodlarındaki işleme tolerans değerleri; 0.3 mm, 0.2 mm, 0.1 mm, 0.05 mm olarak tanımlanmıştır. İşleme toleransına göre oluşturulmuş CNC kodları AutoCAD ortamında okunduğunda kullanıcıya; okunan satır sayısı (adet), toplam takım yolu boyu (mm), en küçük takım yolu boyu (mm), en büyük takım yolu boyu (mm) ve toplam takım sayısı (adet) rapor edilmektedir. Buna göre 0.3 mm işleme toleransına göre çıkarılmış takım yolu şekil 7.26’da gösterilmektedir.



Şekil 7.26. İşlem toleransı 0.3 mm olan takım yolu

Parçanın diğer parametreleri aynı kalmak kaydıyla, farklı işleme toleranslarında işlenmesiyle oluşan takım yolunda, takım yolu boyunun 0.1 mm ile 5 mm arasında gruplandırılarak gösterilmesi Çizelge 7.7’de verilmektedir.

Çizelge 7.11. Farklı işleme toleranslarında takım yollarının, takım yolu boyuna göre gruplandırılması

Takım Yolu Boyu	İşleme Toleransı / Takım Yolu Adedi			
	0.3	0.2	0.1	0.05
$a > 5$	3160	2740	2410	1270
$5 > a > 4,3$	110	630	380	1010
$4,3 > a > 3,6$	449	599	759	659
$3,6 > a > 2,9$	110	370	1190	730
$2,9 > a > 2,2$	524	524	1004	2594
$2,2 > a > 1,5$	48	40	360	1160
$1,5 > a > ,8$	20	28	20	500
$0,8 > a > 0,1$	20	20	28	28
$a < 0,1$	0	0	0	0

Her bir işleme toleransına göre, işleme tolerans değerinden küçük takım yolu sayısı çizelge 7.12’ de verilmektedir.

Çizelge 7.12. Farklı işleme toleranslarında takım yollarının, işleme toleranslarından küçük takım yolları

Takım Yolu Boyu	İşleme Toleransı / Takım Yolu Adedi			
	0.3	0.2	0.1	0.05
$a < 0.3$	0			
$a < 0.2$		0		
$a < 0.1$			0	
$a < 0.05$				0

Çizelgede de görüldüğü gibi işleme toleransından küçük takım yolları oluşmamıştır. Her bir işleme toleransına göre ele alınan takım yollarının bazı bilgileri de Çizelge 7.13’de gösterilmektedir.

Çizelge 7.13. Farklı işleme toleranslarındaki varolan CNC kod sonuçlarının karşılaştırılması

İşleme toleransı (mm)	0.3	0.2	0.1	0.05
Okunan satır sayısı	4451	4961	6161	7961
Toplam takım yolu boyu	68335.62	68353.2	68375.16	68387.98
En büyük takım yolu	216.3785	216.3785	216.3785	216.3785
En küçük takım yolu	0.518	0.518	0.518	0.294
Toplam takım sayısı	1	1	1	1

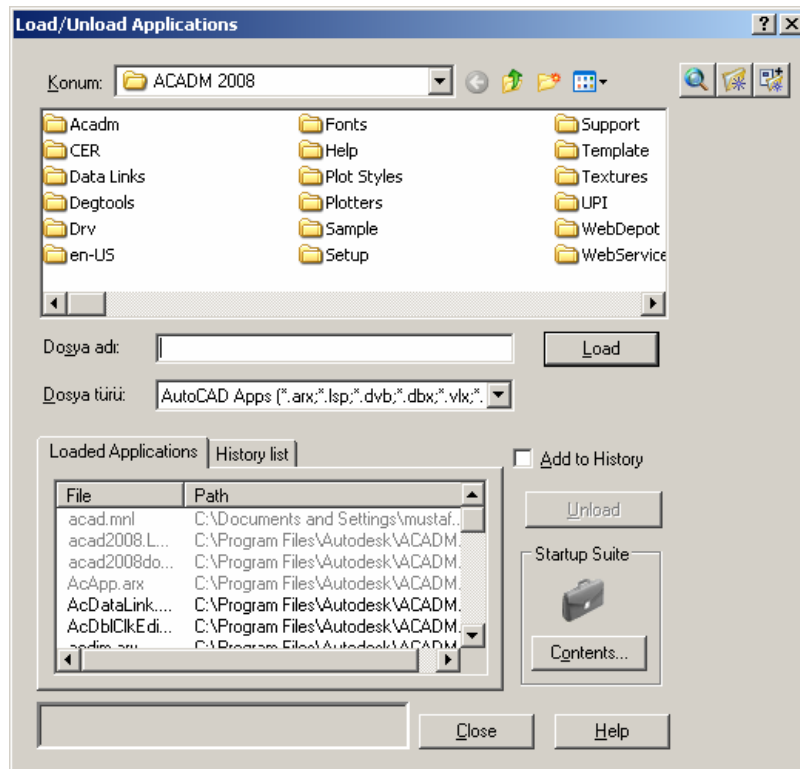
Bu çizelgedeki değerlerden anlaşıldığı gibi işleme toleransı küçüldükçe CNC kodlarındaki satır sayısı yani takım hareketleri artmaktadır. Bu örnek uygulamada işleme toleransının altında takım yolu oluşmadığından takım yolu optimizasyonu yapıldığında çıkarılan takım yolu bulunmamıştır.

8. BİLGİSAYAR DESTEKLİ TAKIM YOLU DOĞRULAMASI VE OPTİMİZASYONU PROGRAMININ KULLANILMASI

Geliştirilen programın kullanılması için AutoCAD 2008 yazılımı kurulu olmalıdır. Ayrıca CNC verilerinin tablo halinde gösterilebilmesi için Microsoft Flex Grid Control, version 6 kontrolünün C:\WINDOWS\system32 klasörüne kaydedilmesi gereklidir.

8.1. Programın Yüklenmesi ve Kullanılması

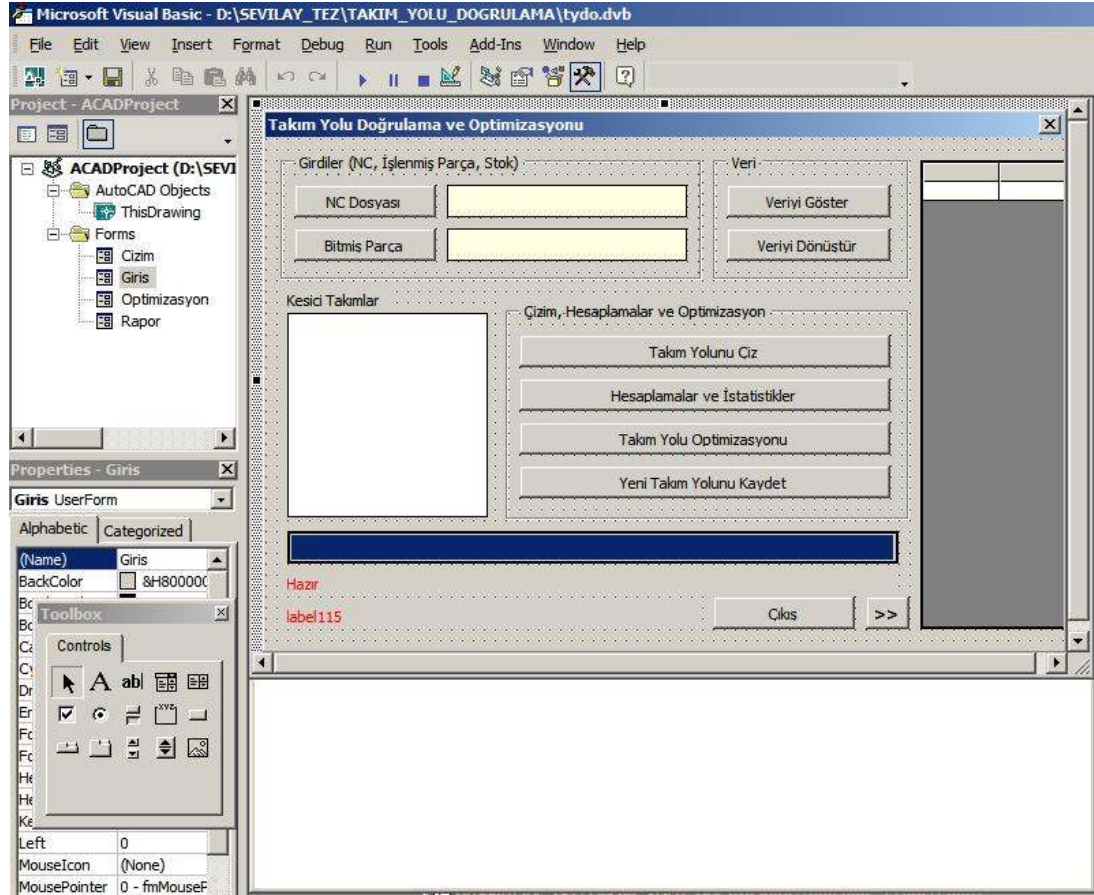
AutoCAD 2008 programı çalıştırıldığında ekrana Şekil 7.1'deki kullanıcı ara yüzü penceresi gelir. AutoCAD düşey menülerinden Tools > Load Application adımları takip edilerek Şekil 8.1'deki "Load/Unload Application" penceresi açılır.



Şekil 8.1. "Load / Unload Application" penceresi

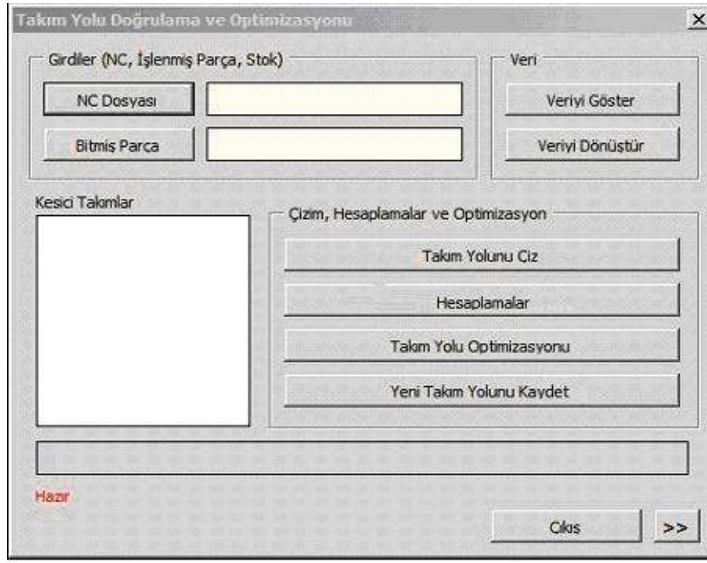
Buradaki "Contents" butonuna tıklayarak Windows gezgini yardımı ile "tydo.dvt" dosyasını seçip yüklenen yardımcı programlara eklenir. Programın eklenmesiyle Visual Basic editörüne program eklenmiş olur. Klavyeden Alt + F11 tuşlarına

basılarak editörü açılır. Şekil 8. 2’de Microsoft Visual Basic 5.0 editörü ve takım yolu doğrulama ve optimizasyonu programı ara yüzü gösterilmektedir.



Şekil 8.2. Microsoft Visual Basic 5.0 editörü ve takım yolu doğrulama ve optimizasyonu programı ara yüzü

Klavyeden F5 tuşuna basılır program çalıştırılır. Program çalıştırılınca Şekil 8.3’deki giriş ara yüzü belirir.



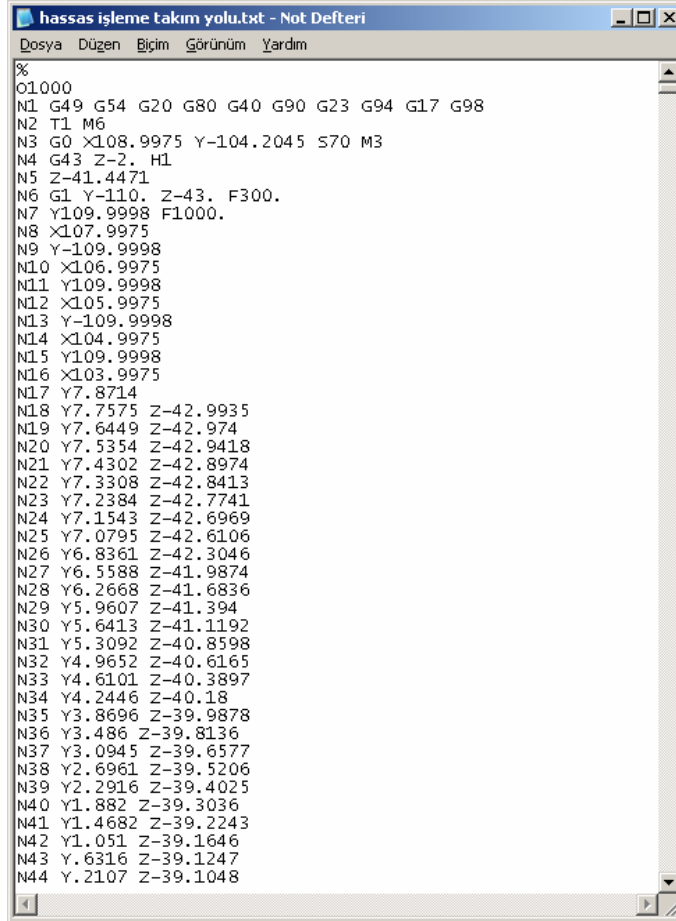
Şekil 8.3. Takım Yolu Doğrulama ve Optimizasyonu program ara yüzü

Program ara yüzünde takım yolu doğrulaması ve optimizasyonu yapılacak parçanın çağrılacağı “CNC Dosyası” butonu en üst sırada bulunmaktadır. Hemen altında aynı parçanın .dxf uzantılı modelini çağıran “Bitmiş Parça” butonu bulunmaktadır. “Veriyi Göster” butonuyla varolan CNC kodları .txt formatında gösterilmektedir. “Veriyi Dönüştür” butonuyla sayısal denetim takım yolları tablo haline getirilir. “Takım Yolunu Çiz” butonuyla takım yolunun takımlara göre, işleme toleransına göre ve hareket tipine göre çizdirilebileceği seçenek butonları açılarak istenilen seçilip çizme işlemi başlatılır. “Hesaplamalar” butonuyla takım yolundaki takım sayısı, okunan satır sayısı, en büyük takım yolu, en küçük takım yolu ve toplam takım yolu boyu değerleri rapor halinde sunulmaktadır. “Takım Yolu Optimizasyonu” butonu takım yolunun işleme toleransına göre optimizasyonunu başlatır. Optimizasyon sonucu oluşan yeni takım yolu da “Yeni Takım Yolunu Kaydet” butonuyla kaydedilir.

8.2. Sayısal Denetim Takım Yolu Verisinin Seçilmesi ve Gösterilmesi

Takım Yolu Doğrulama ve Optimizasyonu penceresinden CNC Dosyası butonuna tıklanarak sayısal denetim veri dosyası seçilir. Dosya seçildiğinde butonun karşısına dosyanın tam yolu ve ismi yazılır. Veri Bölümündeki Veriyi Göster butonuna

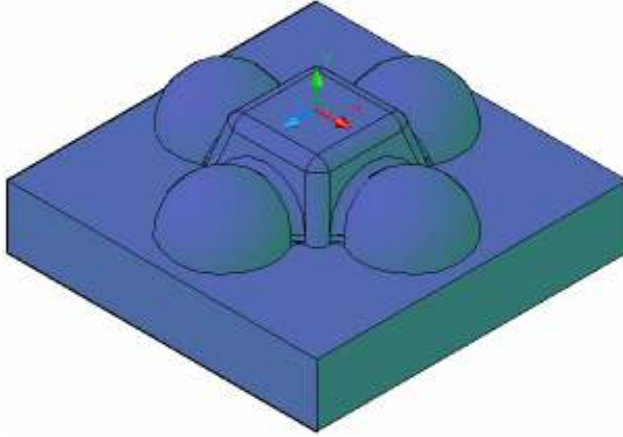
basılarak Not defteri metin editörü yardımı ile sayısal denetim verisi gösterilir (Şekil 8.4).



Şekil 8.4. Sayısal denetim verisinin gösterilmesi

8.3. Bitmiş Parçanın Açılması

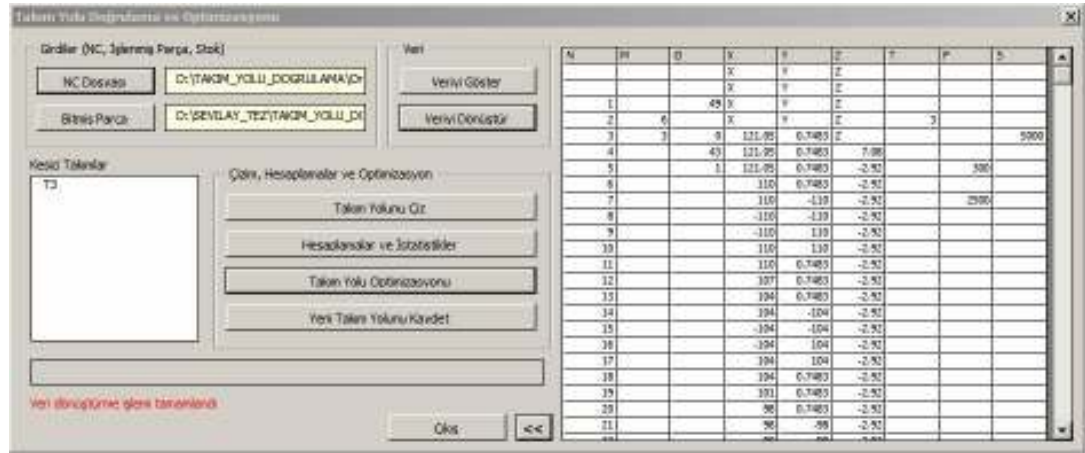
“Takım Yolu Doğrulama ve Optimizasyonu” penceresinden “Bitmiş Parça” butonuna tıklanarak açılan dosya seçim penceresinden optimizasyonu yapılan parça modelinin AutoCAD dosyası seçilir. Dosya seçildiğinde çizim alanına Şekil 8.5’deki gibi çizim alanına eklenecektir.



Şekil 8.5. Bitmiş parçanın AutoCAD çizim alanında gösterilmesi

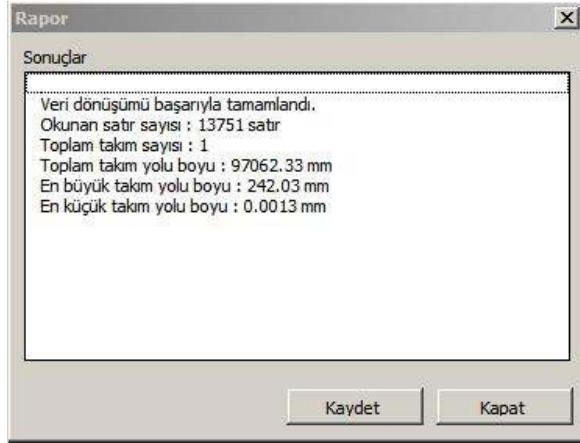
8.4. Sayısal Verinin Dönüştürülmesi

“Takım Yolu Doğrulama ve Optimizasyonu” penceresinden “Veriyi Dönüştür” butonuna tıklanarak seçilen sayısal denetim verisi Şekil 8.6’daki gibi MsFlexGrid tablosuna aktarılacaktır.



Şekil 8.6. Sayısal denetim verisinin dönüştürülmesi

Veri dönüşüm işleminin bitmesi ile beraber takım yolu ile ilgili Şekil 8.7’deki rapor verilecektir. Bu raporda bir önceki bölümde listelenen takım yolu analiz bilgileri verilir. Bu pencereden Kaydet butonu kullanılarak hesaplama sonuçları rapor halinde txt dosyasına kaydedilir.



Şekil 8.7. Takım yolu hesaplama raporu

8.5. Takım Yolunun Çizdirilmesi

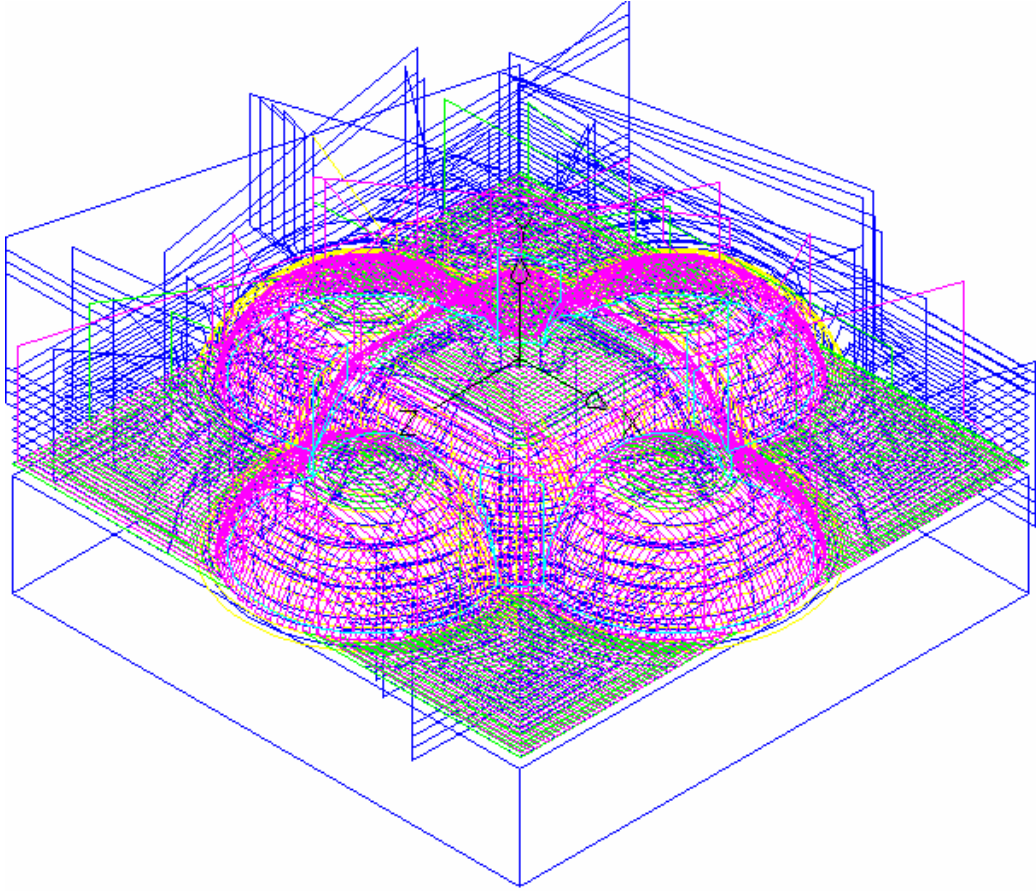
Sayısal denetim verisi dönüştürüldükten sonra “Takım Yolu Doğrulama ve Optimizasyonu” penceresinden “Takım Yolunu Çiz” butonuna tıklanarak Şekil 8.8’de gösterilen “Takım Yolu Çizimi” penceresi açılır.



Şekil 8.8. Takım Yolu Çizim penceresi

8.5.1. Takımlara göre çizim yapılması

Takım yolunun çiziminin ilk yöntemi takımlara göre çizimin yapılmasıdır. Penceredeki “Takımlar” kutucuğu aktifleştirilip “Çiz” butonuna tıklandığında takımlara göre çizim yapılacaktır (Şekil 8.9).



Şekil 8.9. Takımlara göre çizilmiş örnek takım yolu

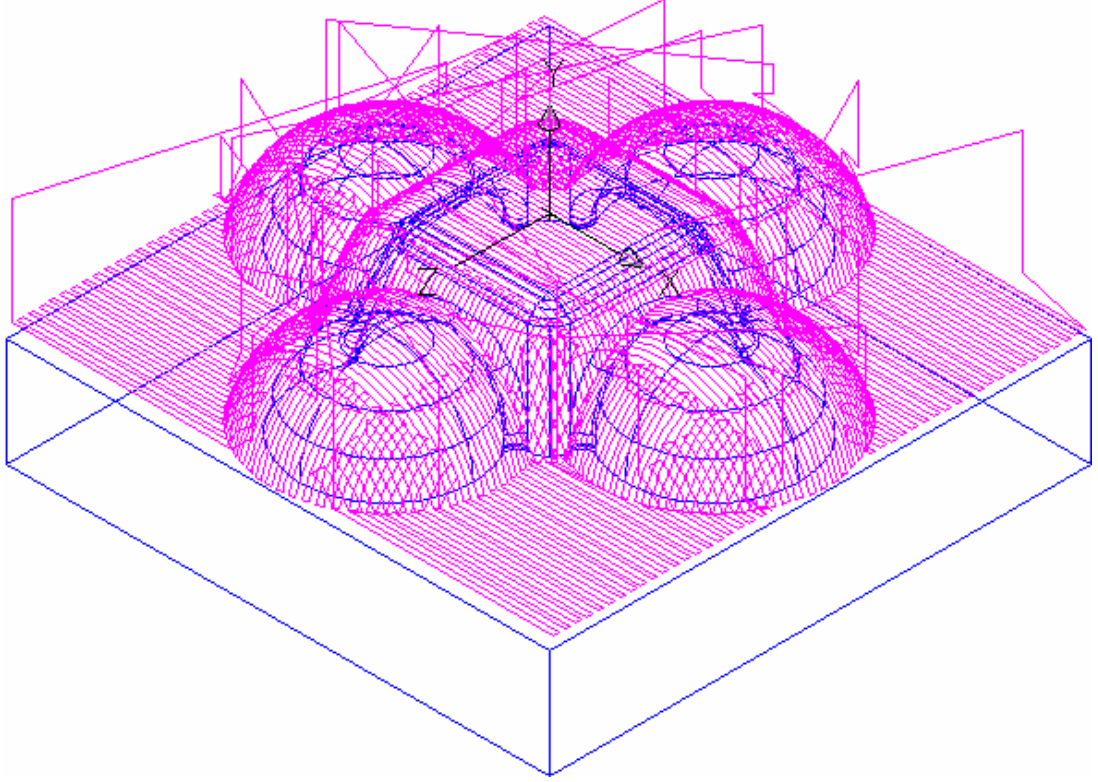
Takımlara göre çizim yapıldığında Şekil 8.10'daki gibi AutoCAD katmanlarına takımlar kaydedilir.



Şekil 8.10. Takım katmanları

Buradan ilgili takımlar AutoCAD yazılımının katmanları yönetim araçlarından Freeze özelliği kullanılarak istenilen takıma ait takım yolları gösterilebilir. Şekil

8.11’de bitirme operasyonunun yapıldığı T4 takım ile işleme takım yolu görülmektedir.



Şekil 8.11. T4 takımı ile işlemenin gösterilmesi

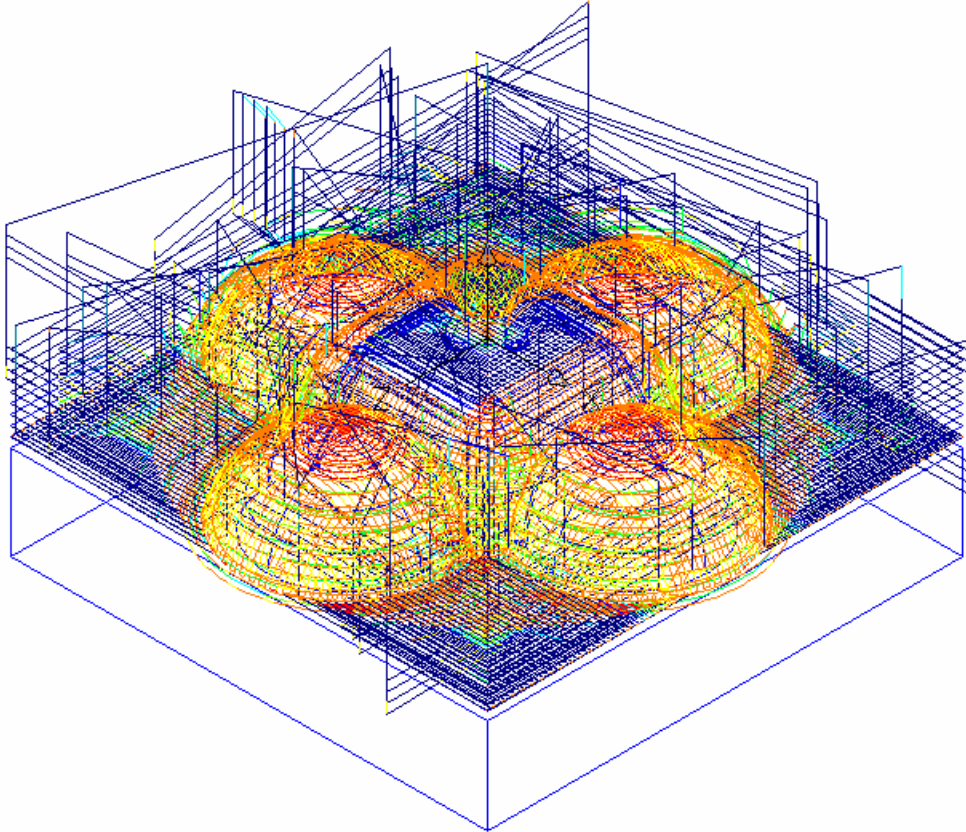
8.5.2. Takım yolu boylarına göre çizim yapılması

“Takım Yolu Çizimi” penceresinden “Takım Yolu Boyu” radio butonu aktifleştirildiğinde pencere Şekil 8.12’deki gibi değişecektir.



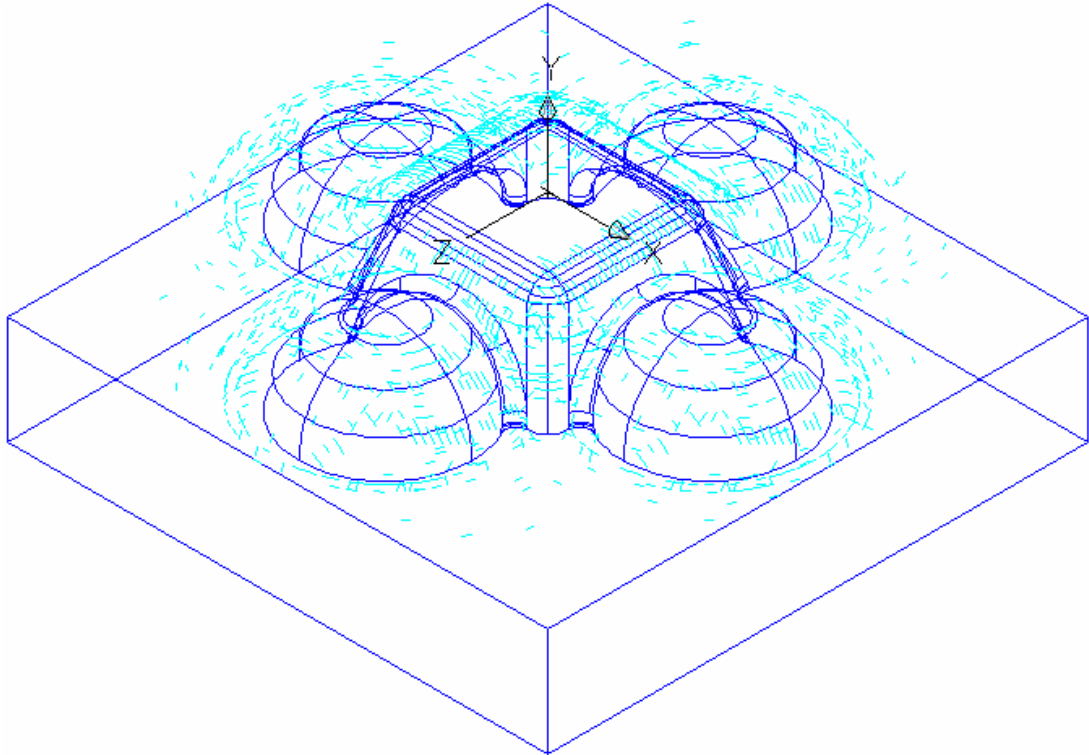
Şekil 8.12. Takım yolu boyuna göre çizim ayarları

Pencereden renk haritası değerleri girilip Çiz butonuna tıklandığında Şekil 8.13'deki gibi çizgi boylarına göre takım yolları çizilecektir.



Şekil 8.13. Takım yolu boyuna göre takım yolunun çizilmesi

Takım yolu boyuna göre takım yolu çizdirildiğinde AutoCAD katmanlarına her takım yolu boyuna göre katmanlar eklenir ve içerlerine takım yolları aktarılır. AutoCAD yazılımının katmanlarının “Freeze” özelliği kullanılarak istenilen takım yolları gizlenip diğerleri üzerinde inceleme yapılır. Şekil 8.14’de 0,54 ve 0,63 arası boyutlardaki takım yolları verilmiştir.



Şekil 8.14. 0,54 – 0,63 arası boyutlardaki takım yollarının gösterilmesi

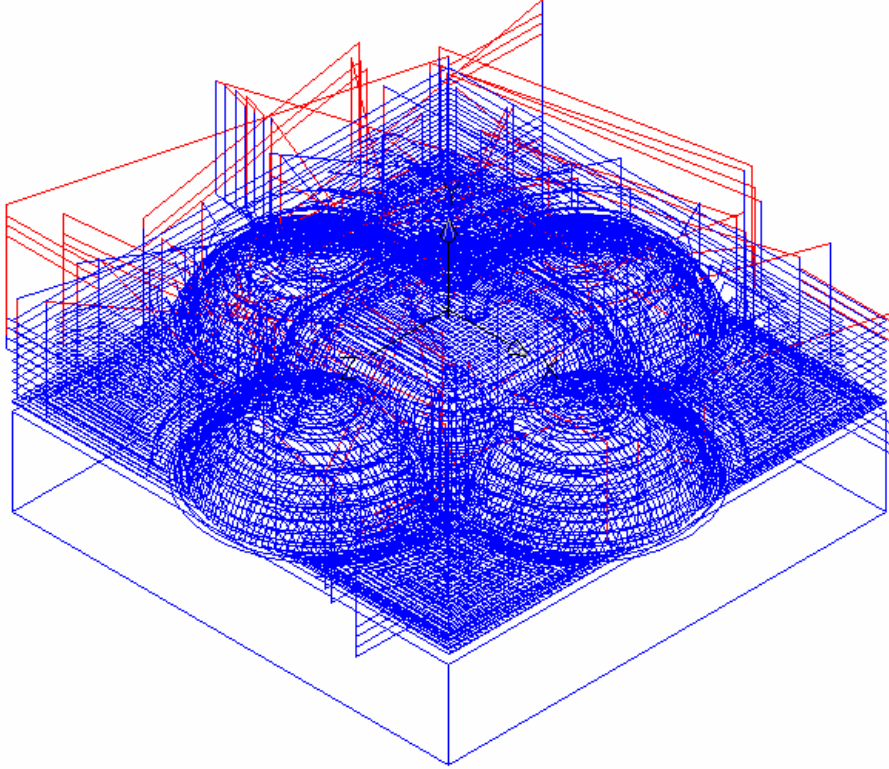
8.5.3. Hareket tipine göre çizim yapılması

“Takım Yolu Çizimi” penceresinden “Hareket Tipi” radio butonu aktifleştirildiğinde pencere Şekil 8.15’deki gibi değişecektir.



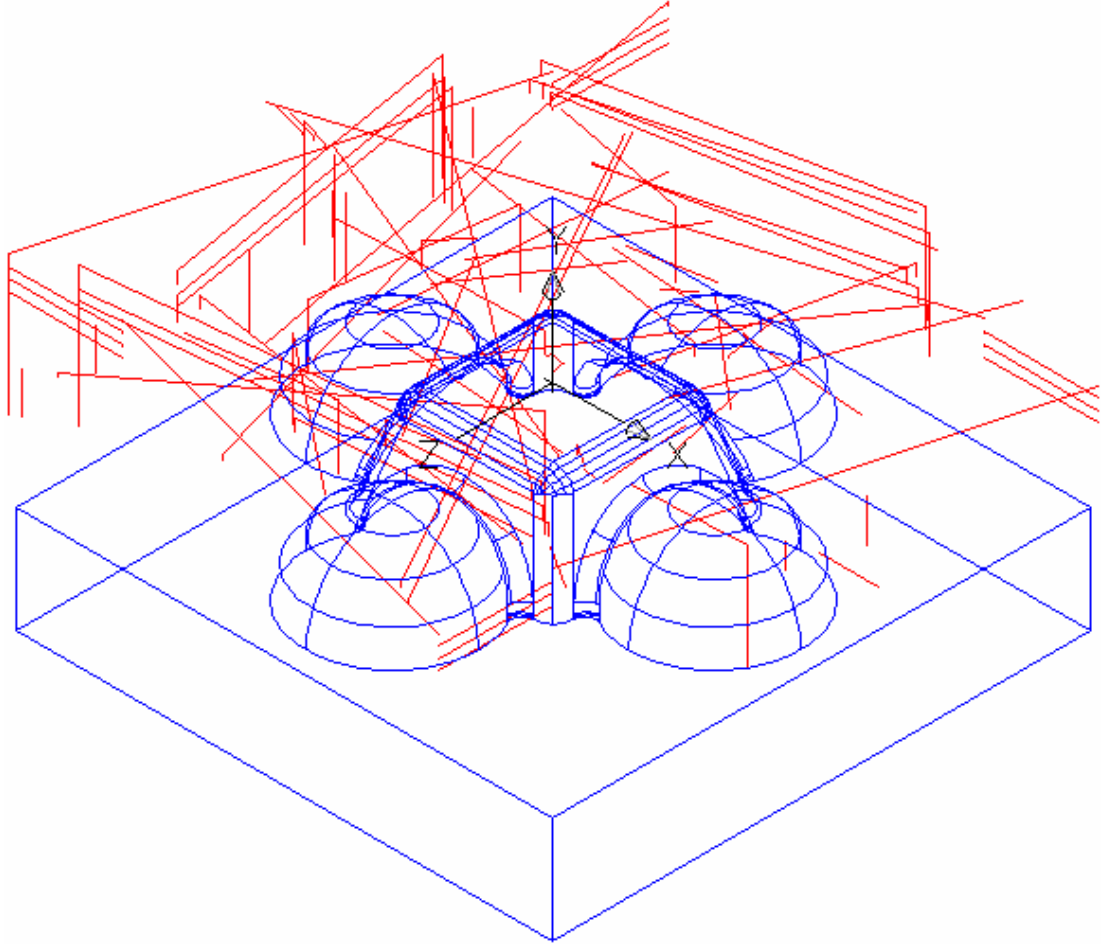
Şekil 8.15. Hareket tipine göre çizim penceresi

Pencereden Çiz butonuna tıklandığında iki renk olarak, talaş kaldırmadan hızlı ilerleme G00 için kırmızı renkler ve talaş kaldırarak ilerleme G01 için mavi renkle takım yolları çizilir (Şekil 8.16).



Şekil 8.16. Hareket tipine göre çizimlerin gösterilmesi

AutoCAD yazılımının Layer katman yöneticisinden istenilen hareket tipi saklanarak diğer hareket tipinde inceleme yapılabilir. Şekil 8.17’de G00 hareketleri görülmektedir.



Şekil 8.17. G00 takım yolu hareketleri

8.6. Hesaplamaların Gösterilmesi

“Takım Yolu Doğrulama ve Optimizasyonu” penceresinden “Hesaplamalar” butonuna tıklandığında takım yolunun, okunan satır sayısı, toplam takım sayısı, toplam takım yolu boyu, en büyük takım yolu boyu, en küçük takım yolu boyu gibi bilgileri rapor eder. (Şekil 8.18).

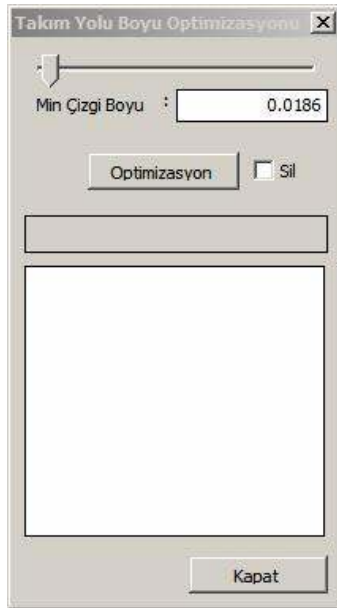


Şekil 8.18. Takım yolu hesaplamaları ve istatistikleri.

Şekil 8.18’deki pencerede “Kaydet” butonuna tıklanıp dosya ismi ve yeri belirlenip kaydedilerek rapor txt formatında kaydedilir. Kaydedilen bu dosyaya istenildiği zaman erişilerek takım yolu ile ilgili bilgiler değerlendirilir.

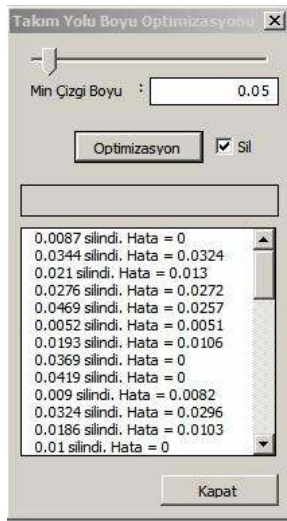
8.7. Takım Yolu Optimizasyonu İşlemi

“Takım Yolu Doğrulama ve Optimizasyonu” penceresinden “Takım Yolu Optimizasyonu” butonuna tıklanarak Şekil 8.19’da verilen takım yolu optimizasyon penceresi açılır.



Şekil 8.19. Takım yolu optimizasyon penceresi.

Şekil 8.19'daki pencereden çizgi boyuna göre optimizasyon gerçekleştirilebilir. “Min. Çizgi Boyu” değer alanına optimizasyon parametresi girilir. Takım yolu optimizasyonu yapılırken oluşan en büyük yüzey hatası değeri de hesaplanarak kullanıcıya rapor halinde sunulur. Optimizasyon butonuna tıklanarak optimizasyon başlatılır. Hesaplama sonucunda belirtilen ölçüden küçük değerler alttaki listede listelenirler (Şekil 8.20).



Şekil 8.20. Takım yolu boyuna göre optimizasyon işlemi

Girilen ölçüden küçük takım yollarını çıkarmak için penceredeki Sil kutucuğu aktifleştirilir ve hesaplama Optimizasyon butonuna tıklanarak hesaplama tekrarlanır. Bulunan satırlar çıkarılacaktır.

9. SONUÇ VE ÖNERİLER

Bu çalışmada VBA yazılımı kullanılarak AutoCAD ara yüzünde CNC kodlarının çizilmesi ve yorumlanmasıyla, CAM programlarından alınan CNC kodların doğrulanması ve optimizasyonu yapılmıştır. Yapılan çalışma ile:

- CNC parça programlarının bir CAM programıyla üretilmesiyle çok küçük takım hareketlerinin çıkarıldığı görülmüştür.
- Çok küçük gibi görülen bu hareketler tezgahta dur/kalklara neden olacağından kesici ve tezgah açısından olumsuzluklara sebebiyet verebilmektedir.
- Özellikle kaba işlemlerde işleme toleransından küçük CAM tarafından çıkarılan kesici hareketler imalat zamanını etkilemektedir.
- Geliştirilen sistemde, varolan çok küçük hareketlerin atılmasıyla oluşturulan yeni optimizasyonlu parça programını daha verimli hale getirmiştir.

Varolan CNC kodlarının işleme toleransına göre optimizasyonu sonucunda CNC satırlarında ve toplam takım yolu boyunda azalma olmuştur. Toplam takım yolu boyu, işleme zamanını etkilediğinden, yapılan bu çalışma sonrası elde edilen CNC programları için işleme zamanının azaldığı görülmektedir.

Bu çalışmaya ilave olarak işleme zamanı hesaplanabilir. Takım hareketlerine göre kaldırılan talaş hacmi hesaplanabilir. Kesici hareket tipine göre ilerleme ve devir hesapları yapılarak değerler kontrol edilebilir.

KAYNAKLAR

1. Kim, S. J., Lee, H. U., Cho, D.W., "Feedrate Scheduling For Indexable End Milling Process Based On An Improved Cutting Force Model" ***Samsung Heavy Industries Co., Ltd.,103-28, Munji-dong, Yuseong-gu, Daejeon*** 305-380, South Korea, ***Department of Mechanical Engineering, Pohang University of Science and Technology, San 31 Hyoja-dong, Nam-gu, Pohang, Kyungbuk*** 790-784, South Korea, Received 22 March 2005 Received In Revised Form 5 September 2005; Accepted 7 September, (2005).
2. Jerard, R. B., Fussell, B. K., Hemmett, J. G., Ercan, M. T. ***Proceedings of the 2000 NSF Design & Manufacturing Research Conference***, "ToolPath Feedrate Optimization: A Case Study", Vancouver, British Columbia, Canada 3-6, (2000).
3. Wang, H., and Stori J. A., "A Metric-Based Approach To 2D Tool-Path Optimization For High-Speed Machining" ***Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign***, Urbana, Illinois, 1-8, (2000).
4. Erdim, H., Lazoglu, I., Ozturk, B., "Feedrate scheduling Strategies For Free-Form surfaces" ***Manufacturing Automation and Research Center, Department of Mechanical Engineering***, Koc University, Sariyer, 34450 Istanbul, Turkey,Received 16 February 2005; accepted 19 July,2-5, (2005).
5. Pateloup, V., Duc, E., Ray, P. "Corner Optimization For Pocket Machining", ***LaRAMA, IFMA***, Campus des Cezeaux, BP 265 63 175 Aubiere Cedex, France Received 1 December 2003; Received in Revised Form 19 April 2004 accepted 26 April, 2-6, (2004).
6. Lazoğlu, I., Manav, C., Murtezaoğlu, Y., "Tool path optimization for free form surface machining", ***Koc University, Manufacturing and Automation Research Center***, Sariyer 34450, Kocaeli, Türkiye, 102-104, (2009).
7. Held, M., Spielberg, C., "A smooth spiral tool path for high speed machining of 2D pockets", ***Universität Salzburg, FB Computerwissenschaften***, A_5020 Salzburg, Avusturya, 540-549, (2009).
8. Oysu, C., Bingül, Z., "Application of heuristic and hybrid-GASA algorithms to tool-path optimization problem for minimizing airtime during Machining", ***Department of Mechatronics Engineering, University of Kocaeli***, Umuttepe Campus, 41380 Kocaeli, Türkiye, 390-395, (2009).
9. Karagülle, İ., Pala Z., "Microsoft Visual Basic 6.0", ***Turkmen Kitabevi***, İstanbul , 12-54, (2001).

10. Başak, H., Editör: Şahin, İ., "VBA ve AutoCAD Uygulamaları", **Asil Yayın Dağıtım Ltd. Şti.**, Ankara, 3-43, (2005).
11. Vatansever, F., "İleri programlama Uygulamaları" **Seçkin Yayıncılık San.Tic.A.Ş.** Ankara, 17-50, (2006).
12. Elber, G., Cohen, E., Drake, S., "MATHSM: Medial axis transform toward high speed machining of pockets", **Computer Aided Design**, 37(2):241_50, 12-25 (2004).
13. Park, S.C., Chung, Y.C., "Offset tool-path linking for pocket machining", **Computer Aided Design**, 34(4):299_308, 30-45, (2002).
14. Oysu, C., Bingul, Z., "Tool path optimization using genetic algorithms", **Conference on Genetic and Evolutionary Methods**, Las Vegas, USA, 120–126, (2007).

EKLER

EK-1 Geliştirilen programın komut satırları

Takımyolu doğrulama ve optimizasyonu için hazırlanan program satırları aşağıda aşama aşama verilmektedir.

1. CNC programı bilgilerinin çıkarılması için hazırlanan program satırları

```
Dim s, satir, satir_s As String
Dim i, ii As Byte
Dim x_yeri, y_yeri, z_yeri, f_yeri, s_yeri, n_yeri, m_yeri, t_yeri, g_yeri As Byte
Dim x_durum, y_durum, z_durum, f_durum, s_durum, n_durum, m_durum, t_durum, g_durum As Boolean
Dim t As Long
Dim x, y, z, f, ss, n, m, g, tt As String
Dim satir_sayisi As Long
Dim PauseTime, Start, Finish, TotalTime As Variant
Dim tek_cif As Byte
Dim liste_satir_sayisi As Integer
Private Type takiml
    takimno As String
    takimcapi As String
    ucradusu As String
    boy As String
    kesmeboyu As String
    tadimtipi As String
    renk As ACAD_COLOR
End Type
Dim pers() As takiml
Dim diziboyutu As Integer
Dim elemansayisi As Integer
Dim j As Double
Dim c1x, c1y, c1z, c2x, c2y, c2z, ektyb, tyb, ekda, toplam_tyb, ort_tyb, geo_otyb, ebtyb As Double
Private Sub CommandButton1_Click()
    CommonDialog1.DialogTitle = "CNC dosyasını seçiniz..!"
    CommonDialog1.Filter = "Metin dosyaları |*.txt|"
    CommonDialog1.ShowOpen
    TextBox4.Text = CommonDialog1.FileName
    Label1.Caption = "CNC verisi seçildi."
End Sub
Private Sub CommandButton12_Click()
    tek_cif = tek_cif + 1
    If (tek_cif Mod 2) = 0 Then
        Giris.Width = 365
        MSFlexGrid1.Visible = False
        CommandButton12.Caption = ">>"
    End If
```

EK-1 (Devam) Geliştirilen programın komut satırları

```

If (tek_cif Mod 2) = 1 Then
Giris.Width = 706
MSFlexGrid1.Visible = True
CommandButton12.Caption = "<<"
End If
End Sub

Private Sub CommandButton13_Click()
pers(ListBox1.ListIndex + 1).takimno = TextBox11.Text
pers(ListBox1.ListIndex + 1).takimcapi = TextBox7.Text
pers(ListBox1.ListIndex + 1).ucradusu = TextBox8.Text
pers(ListBox1.ListIndex + 1).boy = TextBox9.Text
pers(ListBox1.ListIndex + 1).kesmeboyu = TextBox10.Text
If OptionButton1.Value = True Then pers(ListBox1.ListIndex + 1).tadimtipi = 1
If OptionButton2.Value = True Then pers(ListBox1.ListIndex + 1).tadimtipi = 2
If OptionButton3.Value = True Then pers(ListBox1.ListIndex + 1).tadimtipi = 3
ListBox1.List(ListBox1.ListIndex) = ("T" & TextBox11.Text & " Ø" & TextBox7.Text & " R" & TextBox8.Text & " L" &
.Text)
End Sub

Private Sub CommandButton14_Click()
Rapor.Show
End Sub

Private Sub CommandButton2_Click()
Unload Me
End
End Sub

Private Sub xbul()
satir = s
x_yeri = InStr(1, satir, "X")
If Mid(satir, x_yeri + 1, 1) = "-" Or Mid(satir, x_yeri + 1, 1) = "." Then
For i = 1 To 10
x = Mid(satir, x_yeri + 2, i)
x_durum = IsNumeric(x)
If x_durum = False Then Exit For
Next i
MSFlexGrid1.TextMatrix(t, 3) = Val(Mid(satir, x_yeri + 1, i))
End If
If Mid(satir, x_yeri + 1, 2) = "-." Then
For i = 2 To 10
x = Mid(satir, x_yeri + 3, i)
x_durum = IsNumeric(x)
If x_durum = False Then Exit For
Next i
MSFlexGrid1.TextMatrix(t, 3) = Val(Mid(satir, x_yeri + 1, i + 1))
End If
If Mid(satir, x_yeri + 1, 1) Like "#" Then

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

For i = 1 To 10
x = Mid(satir, x_yeri + 1, i)
x_durum = IsNumeric(x)
If x_durum = False Then Exit For
Next i
MSFlexGrid1.TextMatrix(t, 3) = Val(Mid(satir, x_yeri + 1, i - 1))
End If
If MSFlexGrid1.TextMatrix(t, 3) = "" Then
MSFlexGrid1.TextMatrix(t, 3) = MSFlexGrid1.TextMatrix(t - 1, 3)
End If
End Sub
Private Sub ybul()
satir = s
y_yeri = InStr(2, satir, "Y")
If Mid(satir, y_yeri + 1, 1) = "-" Or Mid(satir, y_yeri + 1, 1) = "." Then
For ii = 1 To 10
y = Mid(satir, y_yeri + 2, ii)
y_durum = IsNumeric(y)
If y_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 4) = Val(Mid(satir, y_yeri + 1, ii))
End If
If Mid(satir, y_yeri + 1, 2) = "-." Then
For ii = 2 To 10
y = Mid(satir, y_yeri + 3, ii)
y_durum = IsNumeric(y)
If y_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 4) = Val(Mid(satir, y_yeri + 1, ii + 1))
End If
If Mid(satir, y_yeri + 1, 1) Like "#" Then
For ii = 1 To 10
y = Mid(satir, y_yeri + 1, ii)
y_durum = IsNumeric(y)
If y_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 4) = Val(Mid(satir, y_yeri + 1, ii - 1))
End If
If MSFlexGrid1.TextMatrix(t, 4) = "" Then
MSFlexGrid1.TextMatrix(t, 4) = MSFlexGrid1.TextMatrix(t - 1, 4)
End If
End Sub
Private Sub zbul()
satir = s
z_yeri = InStr(2, satir, "Z")

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

If Mid(satir, z_yeri + 1, 1) = "-" Or Mid(satir, z_yeri + 1, 1) = "." Then
For ii = 1 To 10
z = Mid(satir, z_yeri + 2, ii)
z_durum = IsNumeric(z)
If z_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 5) = Val(Mid(satir, z_yeri + 1, ii))
End If
If Mid(satir, z_yeri + 1, 2) = "-." Then
For ii = 2 To 10
z = Mid(satir, z_yeri + 3, ii)
z_durum = IsNumeric(z)
If z_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 5) = Val(Mid(satir, z_yeri + 1, ii + 1))
End If
If Mid(satir, z_yeri + 1, 1) Like "#" Then
For ii = 1 To 10
z = Mid(satir, z_yeri + 1, ii)
z_durum = IsNumeric(z)
If z_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 5) = Val(Mid(satir, z_yeri + 1, ii - 1))
End If
If MSFlexGrid1.TextMatrix(t, 5) = "" Then
MSFlexGrid1.TextMatrix(t, 5) = MSFlexGrid1.TextMatrix(t - 1, 5)
End If
End Sub

Private Sub CommandButton22_Click()
Optimizasyon.Show
End Sub

Private Sub CommandButton23_Click()
n1 = 0
CommonDialog4.ShowSave
Open CommonDialog4.FileName For Append As #3
Print #3, "%"
Print #3, "1000"
Close #3
For i = 1 To MSFlexGrid1.Rows - 1
If MSFlexGrid1.TextMatrix(i, 0) <> "" Then
Open CommonDialog4.FileName For Append As #3
If MSFlexGrid1.TextMatrix(i, 0) <> "" Then
n_satiri = "N" & i - 2
End If
If MSFlexGrid1.TextMatrix(i, 0) = "" Then n_satiri = ""

```


EK-1 (Devam) Geliştirilen programın komut satırları

```

If MSFlexGrid1.TextMatrix(i, 1) <> "" Then m_satir = " M" & MSFlexGrid1.TextMatrix(i, 1)
If MSFlexGrid1.TextMatrix(i, 1) = "" Then m_satir = ""
If MSFlexGrid1.TextMatrix(i, 2) <> "" Then g_satir = " G" & MSFlexGrid1.TextMatrix(i, 2)
If MSFlexGrid1.TextMatrix(i, 2) = "" Then g_satir = ""
If (IsNumeric(MSFlexGrid1.TextMatrix(i, 3)) = True) And (MSFlexGrid1.TextMatrix(i, 3) <> MSFlexGrid1.TextMatrix(i - 1,
)) Then x_satir = " X" & MSFlexGrid1.TextMatrix(i, 3)
If MSFlexGrid1.TextMatrix(i, 3) = MSFlexGrid1.TextMatrix(i - 1, 3) Then x_satir = ""
If (IsNumeric(MSFlexGrid1.TextMatrix(i, 4)) = True) And (MSFlexGrid1.TextMatrix(i, 4) <> MSFlexGrid1.TextMatrix(i - 1,
)) Then y_satir = " Y" & MSFlexGrid1.TextMatrix(i, 4)
If MSFlexGrid1.TextMatrix(i, 4) = MSFlexGrid1.TextMatrix(i - 1, 4) Then y_satir = ""
If (IsNumeric(MSFlexGrid1.TextMatrix(i, 5)) = True) And (MSFlexGrid1.TextMatrix(i, 5) <> MSFlexGrid1.TextMatrix(i - 1,
)) Then z_satir = " Z" & MSFlexGrid1.TextMatrix(i, 5)
If MSFlexGrid1.TextMatrix(i, 5) = MSFlexGrid1.TextMatrix(i - 1, 5) Then z_satir = ""
If MSFlexGrid1.TextMatrix(i, 6) <> "" Then t_satir = " T" & MSFlexGrid1.TextMatrix(i, 6)
If MSFlexGrid1.TextMatrix(i, 6) = "" Then t_satir = ""
If MSFlexGrid1.TextMatrix(i, 7) <> "" Then f_satir = " F" & MSFlexGrid1.TextMatrix(i, 7)
If MSFlexGrid1.TextMatrix(i, 7) = "" Then f_satir = ""
If MSFlexGrid1.TextMatrix(i, 8) <> "" Then s_satir = " S" & MSFlexGrid1.TextMatrix(i, 8)
If MSFlexGrid1.TextMatrix(i, 8) = "" Then s_satir = ""
satir = n_satir & m_satir & g_satir & x_satir & y_satir & z_satir & t_satir & f_satir & s_satir
Print #3, satir
Close #3
End If
Next i
Open CommonDialog4.FileName For Append As #3
Print #3, "%"
Close #3
End Sub

Private Sub CommandButton3_Click()
Giris.Hide
Cizim.Show
End Sub

Private Sub gbul()
satir = s
g_yeri = InStr(1, satir, "G")
If Mid(satir, g_yeri + 1, 1) Like "#" Then
For ii = 1 To 3
g = Mid(satir, g_yeri + 1, ii)
g_durum = IsNumeric(g)
If g_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 2) = Val(Mid(satir, g_yeri + 1, ii - 1))
End If
End Sub

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

Private Sub takım_bul()
t_yeri = InStr(1, s, "T")
If Mid(s, t_yeri + 1, 1) Like "#" Then
For ii = 1 To 3
tt = Mid(s, t_yeri + 1, ii)
t_durum = IsNumeric(tt)
If t_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 6) = Val(Mid(s, t_yeri + 1, ii - 1))
ListBox1.AddItem ("T" & Val(Mid(s, t_yeri + 1, ii - 1)))
End If
End Sub

Private Sub fbul()
satir = s
f_yeri = InStr(1, satir, "F")
If Mid(satir, f_yeri + 1, 1) Like "#" Then
For ii = 1 To 6
f = Mid(satir, f_yeri + 1, ii)
f_durum = IsNumeric(f)
If f_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 7) = Val(Mid(satir, f_yeri + 1, ii - 1))
End If
End Sub

Private Sub sbul()
satir = s
s_yeri = InStr(1, satir, "S")
If Mid(satir, s_yeri + 1, 1) Like "#" Then
For ii = 1 To 6
ss = Mid(satir, s_yeri + 1, ii)
s_durum = IsNumeric(ss)
If s_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 8) = Val(Mid(satir, s_yeri + 1, ii - 1))
End If
End Sub

Private Sub nbul()
satir = s
n_yeri = InStr(1, satir, "N")
If Mid(satir, n_yeri + 1, 1) Like "#" Then
For ii = 1 To 7
n = Mid(satir, n_yeri + 1, ii)
n_durum = IsNumeric(n)
If n_durum = False Then Exit For
Next ii

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

MSFlexGrid1.TextMatrix(t, 0) = Val(Mid(satir, n_yeri + 1, ii - 1))
End If
End Sub
Private Sub mbul()
satir = s
m_yeri = InStr(1, satir, "M")
If Mid(satir, m_yeri + 1, 1) Like "#" Then
For ii = 1 To 3
m = Mid(satir, m_yeri + 1, ii)
m_durum = IsNumeric(m)
If m_durum = False Then Exit For
Next ii
MSFlexGrid1.TextMatrix(t, 1) = Val(Mid(satir, m_yeri + 1, ii - 1))
End If
End Sub
Private Sub CommandButton5_Click()
CommonDialog3.DialogTitle = "İşlenmiş paça dosyasını seçiniz..!"
CommonDialog3.Filter = "AutoCAD dosyası |*.dwg|"
CommonDialog3.ShowOpen
TextBox6.Text = CommonDialog3.FileName
If TextBox6.Text = "" Then Exit Sub
ThisDrawing.Application.Documents.Open (CommonDialog3.FileName)
ZoomAll
ThisDrawing.SendCommand "shademode c "
Label1.Caption = "Bitmiş parça oluşturuldu"
End Sub
Private Sub CommandButton6_Click()
ektyb = 100
tyb = 100
Open TextBox4.Text For Input As #1
Label1.Caption = "Dosya verisi inceleniyor"
While Not EOF(1)
Line Input #1, s
satir_sayisi = satir_sayisi + 1
DoEvents
Wend
Close #1
Open TextBox4.Text For Input As #1
t = 0
While Not EOF(1)
Line Input #1, s
ProgressBar1.Max = satir_sayisi
ProgressBar1.Value = ProgressBar1.Value + 1
MSFlexGrid1.Rows = MSFlexGrid1.Rows + 1
t = t + 1

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

Label1.Caption = "Veri dönüştürülüyor. Toplam satır sayısı = " & satir_sayisi & ". Okunan Satır numarası = " & t
Label115.Caption = s
takim_bul
xbul
ybul
zbul
fbul
sbul
nbul
mbul
gbul
DoEvents
Wend
Close #1
Label115.Caption = ""
Label1.Caption = "CNC verisi inceleniyor"
ProgressBar1.Value = 0
On Local Error GoTo 10
For i = 0 To MSFlexGrid1.Rows - 1
If (IsNumeric(MSFlexGrid1.TextMatrix(i, 3)) = True) And (IsNumeric(MSFlexGrid1.TextMatrix(i, 4)) = True) And
IsNumeric(MSFlexGrid1.TextMatrix(i, 5)) = True) Then
c1x = MSFlexGrid1.TextMatrix(i, 3)
c1y = MSFlexGrid1.TextMatrix(i, 4)
c1z = MSFlexGrid1.TextMatrix(i, 5)
c2x = MSFlexGrid1.TextMatrix(i + 1, 3)
c2y = MSFlexGrid1.TextMatrix(i + 1, 4)
c2z = MSFlexGrid1.TextMatrix(i + 1, 5)
tyb = Round((((c2x - c1x) ^ 2 + (c2y - c1y) ^ 2 + (c2z - c1z) ^ 2) ^ 0.5), 4)
DoEvents
ProgressBar1.Value = i
If tyb <> 0 Then
j = j + 1
If tyb < ektyb Then ektyb = tyb
If tyb > ebtyb Then ebtyb = tyb
toplam_tyb = toplam_tyb + tyb
ort_tyb = toplam_tyb / j
End If
End If
Next i
10
Label1.Caption = "Veri dönüştürme işlemi tamamlandı"
ProgressBar1.Value = 0
Rapor.ListBox1.AddItem ("_____")
Rapor.ListBox1.AddItem ("Veri dönüşümü başarıyla tamamlandı.")
Rapor.ListBox1.AddItem ("Okunan satır sayısı : " & satir_sayisi & " satır")

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

Rapor.ListBox1.AddItem ("Toplam takım sayısı : " & ListBox1.ListCount)
'Rapor.ListBox1.AddItem ("G00 hareket boyu : " & 0 & " mm")
'Rapor.ListBox1.AddItem ("G01 hareket boyu : " & 0 & " mm")
'Rapor.ListBox1.AddItem ("Toplam işleme zamanı : " & 0 & " dak")
'Rapor.ListBox1.AddItem ("G00 işleme zamanı : " & 0 & " dak")
'Rapor.ListBox1.AddItem ("G01 işleme zamanı : " & 0 & " dak")
Rapor.ListBox1.AddItem ("Toplam takım yolu boyu : " & Round(toplam_tyb, 2) & " mm")
Rapor.ListBox1.AddItem ("En büyük takım yolu boyu : " & ebtyb & " mm")
Rapor.ListBox1.AddItem ("En küçük takım yolu boyu : " & ektyb & " mm")
Cizim.Label17.Caption = Round(ektyb, 4)
Cizim.Label3.Caption = Round(ort_tyb, 4)
'Rapor.ListBox1.AddItem ("Ortalama takım yolu boyu : " & Round(ort_tyb, 2) & " mm")
'Rapor.ListBox1.AddItem ("En küçük dönüş açısı : " & ekda & " mm")
Rapor.Show
End Sub

Private Sub CommandButton8_Click()
Dim program
program = Shell("notepad.exe", vbNormalFocus)
SendKeys "%da", -1
SendKeys TextBox4.Text, -1
SendKeys "{ENTER}"
Label1.Caption = "Veri başarıyla gösterildi"
End Sub

Private Sub Label1_Click()
Userform5.Show
End Sub

Private Sub UserForm_Activate()
Dim satir_genisligi As Integer
For satir_genisligi = 0 To 8
MSFlexGrid1.ColWidth(satir_genisligi) = 698
Next satir_genisligi
MSFlexGrid1.TextMatrix(0, 0) = "N"
MSFlexGrid1.TextMatrix(0, 1) = "M"
MSFlexGrid1.TextMatrix(0, 2) = "G"
MSFlexGrid1.TextMatrix(0, 3) = "X"
MSFlexGrid1.TextMatrix(0, 4) = "Y"
MSFlexGrid1.TextMatrix(0, 5) = "Z"
MSFlexGrid1.TextMatrix(0, 6) = "T"
MSFlexGrid1.TextMatrix(0, 7) = "F"
MSFlexGrid1.TextMatrix(0, 8) = "S"
Label15.Caption = ""
End Sub

Private Sub UserForm_Initialize()
Giris.Width = 365
MSFlexGrid1.Visible = False

```

EK-1 (Devam) Geliştirilen programın komut satırları

```
diziboyutu = 50
elemansayisi = 0
ReDim pers(diziboyutu)
End Sub
```

2. Kesici hareketlerinin AutoCAD ortamında çizdirilmesi

```
Option Explicit
Dim deger As Double
Dim l1, l2, l3, aci As Double
Dim dosya_adi As String
Dim toplam As Double
Dim katman_adi_sayaci As Byte
Dim cizgi_kalinligi_sayaci, cizgi_aci_sayaci As Byte
Dim takim_var As Boolean
Dim g_var As Boolean
Dim g_durum As Boolean
Dim kalinlik_var, aci_var As Boolean
Dim takim_katmanlari As AcadLayer
Dim g_katmanlari As AcadLayer
Dim kalinlik_katmanlari, aci_katmanlari As AcadLayer
Dim baslangic_noktasi(0 To 2) As Double
Dim bitis_noktasi(0 To 2) As Double
Dim ikinci_bitis_noktasi(0 To 2) As Double
Dim cizgi As AcadLine
Dim k As Long
Dim katmanlari As Variant
Private Sub CommandButton1_Click()
Cizim.Hide
Giris.Show
End Sub
Private Sub CommandButton4_Click()
If OptionButton1.Value = True Then
ProgressBar1.Max = Giris.MSFlexGrid1.Rows
katman_adi_sayaci = 0
ProgressBar1.Value = 0
takim_var = False
For k = 1 To Giris.MSFlexGrid1.Rows - 3
ProgressBar1.Value = ProgressBar1.Value + 1
DoEvents
If Giris.MSFlexGrid1.TextMatrix(k, 6) <> "" Then
takim_var = True
katman_adi_sayaci = katman_adi_sayaci + 1
Set takim_katmanlari = ThisDrawing.Layers.Add("T" & katman_adi_sayaci)
```

EK-1 (Devam) Geliştirilen programın komut satırları

```

If katman_adi_sayaci = 0 Then takim_katmanlari.color = acRed
If katman_adi_sayaci = 1 Then takim_katmanlari.color = acBlue
If katman_adi_sayaci = 2 Then takim_katmanlari.color = acYellow
If katman_adi_sayaci = 3 Then takim_katmanlari.color = acGreen
If katman_adi_sayaci = 4 Then takim_katmanlari.color = acMagenta
If katman_adi_sayaci = 5 Then takim_katmanlari.color = acCyan
End If

If IsNumeric(Giris.MSFlexGrid1.TextMatrix(k, 3)) And IsNumeric(Giris.MSFlexGrid1.TextMatrix(k, 4)) And
(Giris.MSFlexGrid1.TextMatrix(k, 5)) Then
    baslangic_noktasi(0) = Giris.MSFlexGrid1.TextMatrix(k, 3)
    baslangic_noktasi(1) = Giris.MSFlexGrid1.TextMatrix(k, 4)
    baslangic_noktasi(2) = Giris.MSFlexGrid1.TextMatrix(k, 5)
    bitis_noktasi(0) = Giris.MSFlexGrid1.TextMatrix(k + 1, 3)
    bitis_noktasi(1) = Giris.MSFlexGrid1.TextMatrix(k + 1, 4)
    bitis_noktasi(2) = Giris.MSFlexGrid1.TextMatrix(k + 1, 5)
    Set cizgi = ThisDrawing.ModelSpace.AddLine(baslangic_noktasi, bitis_noktasi)
    If takim_var = True Then
        cizgi.Layer = "T" & katman_adi_sayaci
    End If
    cizgi.Layer = "T" & katman_adi_sayaci
    cizgi.Update
End If
Next k

Label1.Caption = "Takım yolu başarıyla çizdirildi"
ProgressBar1.Value = 0
End If

If OptionButton2.Value = True Then
    ProgressBar1.Max = Giris.MSFlexGrid1.Rows
    cizgi_kalinligi_sayaci = 0
    ProgressBar1.Value = 0
    kalinlik_var = False
    ThisDrawing.SendCommand "shademode c "
    ZoomAll
    Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K1")
    kalinlik_katmanlari.color = 174
    Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K2")
    kalinlik_katmanlari.color = 170
    Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K3")
    kalinlik_katmanlari.color = 142
    Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K4")
    kalinlik_katmanlari.color = 130
    Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K5")
    kalinlik_katmanlari.color = 100
    Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K6")
    kalinlik_katmanlari.color = 70

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K7")
kalinlik_katmanlari.color = 50
Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K8")
kalinlik_katmanlari.color = 30
Set kalinlik_katmanlari = ThisDrawing.Layers.Add("K9")
kalinlik_katmanlari.color = 10
For k = 1 To Giris.MSFlexGrid1.Rows - 3
ProgressBar1.Value = ProgressBar1.Value + 1
DoEvents
If IsNumeric(Giris.MSFlexGrid1.TextMatrix(k, 3)) And IsNumeric(Giris.MSFlexGrid1.TextMatrix(k, 4)) And
(Giris.MSFlexGrid1.TextMatrix(k, 5)) Then
baslangic_noktasi(0) = Giris.MSFlexGrid1.TextMatrix(k, 3)
baslangic_noktasi(1) = Giris.MSFlexGrid1.TextMatrix(k, 4)
baslangic_noktasi(2) = Giris.MSFlexGrid1.TextMatrix(k, 5)
bitis_noktasi(0) = Giris.MSFlexGrid1.TextMatrix(k + 1, 3)
bitis_noktasi(1) = Giris.MSFlexGrid1.TextMatrix(k + 1, 4)
bitis_noktasi(2) = Giris.MSFlexGrid1.TextMatrix(k + 1, 5)
Set cizgi = ThisDrawing.ModelSpace.AddLine(baslangic_noktasi, bitis_noktasi)
If (cizgi.Length > Label3.Caption) Then cizgi.Layer = "K1"
If (cizgi.Length > Label5.Caption) And (cizgi.Length <= Label3.Caption) Then cizgi.Layer = "K2"
If (cizgi.Length > Label7.Caption) And (cizgi.Length <= Label5.Caption) Then cizgi.Layer = "K3"
If (cizgi.Length > Label9.Caption) And (cizgi.Length <= Label7.Caption) Then cizgi.Layer = "K4"
If (cizgi.Length > Label11.Caption) And (cizgi.Length <= Label9.Caption) Then cizgi.Layer = "K5"
If (cizgi.Length > Label13.Caption) And (cizgi.Length <= Label11.Caption) Then cizgi.Layer = "K6"
If (cizgi.Length > Label15.Caption) And (cizgi.Length <= Label13.Caption) Then cizgi.Layer = "K7"
If (cizgi.Length > Label17.Caption) And (cizgi.Length <= Label15.Caption) Then cizgi.Layer = "K8"
If cizgi.Length <= Label17.Caption Then cizgi.Layer = "K9"
cizgi.Update
End If
Next k
Label1.Caption = "Takım yolu başarıyla çizdirildi"
ProgressBar1.Value = 0
End If
If OptionButton5.Value = True Then
ProgressBar1.Max = Giris.MSFlexGrid1.Rows
ProgressBar1.Value = 0
g_var = False
ThisDrawing.SendCommand "shademode c "
ZoomAll
Set g_katmanlari = ThisDrawing.Layers.Add("G00")
g_katmanlari.color = 100
Set g_katmanlari = ThisDrawing.Layers.Add("G01")
g_katmanlari.color = 200
For k = 1 To Giris.MSFlexGrid1.Rows - 3
If Giris.MSFlexGrid1.TextMatrix(k, 2) = "0" Then g_durum = True

```


EK-1 (Devam) Geliştirilen programın komut satırları

```

If Giris.MSFlexGrid1.TextMatrix(k, 2) = "1" Then g_durum = False
ProgressBar1.Value = ProgressBar1.Value + 1
DoEvents
If IsNumeric(Giris.MSFlexGrid1.TextMatrix(k, 3)) And IsNumeric(Giris.MSFlexGrid1.TextMatrix(k, 4)) And
(Giris.MSFlexGrid1.TextMatrix(k, 5)) Then
baslangic_noktasi(0) = Giris.MSFlexGrid1.TextMatrix(k, 3)
baslangic_noktasi(1) = Giris.MSFlexGrid1.TextMatrix(k, 4)
baslangic_noktasi(2) = Giris.MSFlexGrid1.TextMatrix(k, 5)
bitis_noktasi(0) = Giris.MSFlexGrid1.TextMatrix(k + 1, 3)
bitis_noktasi(1) = Giris.MSFlexGrid1.TextMatrix(k + 1, 4)
bitis_noktasi(2) = Giris.MSFlexGrid1.TextMatrix(k + 1, 5)
Set cizgi = ThisDrawing.ModelSpace.AddLine(baslangic_noktasi, bitis_noktasi)
If g_durum = True Then cizgi.Layer = "G00"
If g_durum = False Then cizgi.Layer = "G01"
cizgi.Update
End If
Next k
End If
End Sub
Private Sub Frame4_Click()
End Sub
Private Sub Label10_Click()
End Sub
Private Sub Label3_Click()
deger = InputBox("Değer girin.")
Label3.Caption = deger
Label5.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 6), 2)
Label7.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 5), 2)
Label9.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 4), 2)
Label11.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 3), 2)
Label13.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 2), 2)
Label15.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 1), 2)
End Sub
Private Sub Label17_Click()
deger = InputBox("Değer girin.")
Label17.Caption = deger
Label5.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 6), 2)
Label7.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 5), 2)
Label9.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 4), 2)
Label11.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 3), 2)
Label13.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 2), 2)
Label15.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 1), 2)
End Sub
Private Sub OptionButton1_Click()
Gizle

```

EK-1 (Devam) Geliştirilen programın komut satırları

```

End Sub
Private Sub OptionButton2_Click()
    gizle
    If OptionButton2.Value = True Then
        Label5.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 6), 4)
        Label7.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 5), 4)
        Label9.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 4), 4)
        Label11.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 3), 4)
        Label13.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 2), 4)
        Label15.Caption = Round((Label17.Caption + (Label3.Caption - Label17.Caption) / 7 * 1), 4)
    End If
    If OptionButton2.Value = True Then Frame4.Visible = True
End Sub
Private Sub OptionButton5_Click()
    gizle
End Sub
Private Sub OptionButton6_Click()
    gizle
End Sub
Private Sub OptionButton7_Click()
    gizle
End Sub
Private Sub gizle()
    Frame4.Visible = False
End Sub
Private Sub ProgressBar1_MouseDown(ByVal Button As Integer, ByVal Shift As Integer, ByVal x As
.OLE_XPOS_PIXELS, ByVal y As stdole.OLE_YPOS_PIXELS)
End Sub
Private Sub UserForm_Activate()
    If OptionButton1.Value = True Then Frame4.Visible = False
End Sub
Private Sub CommandButton3_Click()
    CommonDialog1.ShowSave
    dosya_adi = CommonDialog1.FileName
    ThisDrawing.SaveAs (dosya_adi)
End Sub

```

Takımyolu doğrulama ve optimizasyonu için varolan CNC kodlarının rapor edilmesi hazırlanan program satırları

```

Option Explicit
Dim s As String
Dim i As Double

```

EK-1 (Devam) Geliştirilen programın komut satırları

```
Private Sub CommandButton2_Click()
    CommonDialog3.ShowSave
    For i = 0 To ListBox1.ListCount - 1
        Open CommonDialog3.FileName For Append As #1
        s = ListBox1.List(i)
        Print #1, s
    DoEvents
    Close #1
    Next i
End Sub

Private Sub ListBox1_Click()
End Sub

Private Sub UserForm_Click()
End Sub

Private Sub CommandButton1_Click()
    Me.Hide
End Sub
```

Takımyolu doğrulama ve optimizasyonu için varolan CNC kodlarının optimizasyonu hazırlanan program satırları:

```
Dim i, j, k, l, b, p, c, alfa, hata, c1x, c1y, c1z, c2x, c2y, c2z, c3x, c3y, c3z, en_buyuk_hata, toplam_hatalar, ortalama_hata As
Sub CommandButton1_Click()
    ListBox1.Clear
    ortalama_hata = 0
    toplam_hatalar = 0
    en_buyuk_hata = 0
    ProgressBar1.Max = Giris.MSFlexGrid1.Rows - 1
    For i = 0 To Giris.MSFlexGrid1.Rows - 1
        If (IsNumeric(Giris.MSFlexGrid1.TextMatrix(i, 3)) = True) And (IsNumeric(Giris.MSFlexGrid1.TextMatrix(i, 4)) = True) And
            IsNumeric(Giris.MSFlexGrid1.TextMatrix(i, 5)) = True) Then
            ProgressBar1.Value = ProgressBar1.Value + 1
        DoEvents
        c1x = Giris.MSFlexGrid1.TextMatrix(i, 3)
        c1y = Giris.MSFlexGrid1.TextMatrix(i, 4)
        c1z = Giris.MSFlexGrid1.TextMatrix(i, 5)
        c2x = Giris.MSFlexGrid1.TextMatrix(i + 1, 3)
        c2y = Giris.MSFlexGrid1.TextMatrix(i + 1, 4)
        c2z = Giris.MSFlexGrid1.TextMatrix(i + 1, 5)
        c3x = Giris.MSFlexGrid1.TextMatrix(i + 2, 3)
        c3y = Giris.MSFlexGrid1.TextMatrix(i + 2, 4)
        c3z = Giris.MSFlexGrid1.TextMatrix(i + 2, 5)
        On Local Error GoTo 10
```

EK-1 (Devam) Geliştirilen programın komut satırları

```

b = Round((((c2x - c1x) ^ 2 + (c2y - c1y) ^ 2 + (c2z - c1z) ^ 2) ^ 0.5), 4)
p = Round((((c3x - c2x) ^ 2 + (c3y - c2y) ^ 2 + (c3z - c2z) ^ 2) ^ 0.5), 4)
c = Round((((c3x - c1x) ^ 2 + (c3y - c1y) ^ 2 + (c3z - c1z) ^ 2) ^ 0.5), 4)
If (Val(c * 10000) < (Val(b * 10000) + Val(p * 10000))) And (Val(b * 10000) < (Val(c * 10000) + Val(p * 10000))) And
Val(p * 10000) < (Val(c * 10000) + Val(b * 10000))) Then
alfa = 90 - Atn((((b ^ 2 + c ^ 2 - p ^ 2) / (2 * b * c)) / ((b ^ 2 + c ^ 2 - p ^ 2) / (2 * b * c)) ^ 2 + 1) ^ 0.5) / (3.1415 / 180)
k = Cos(alfa * (3.1415 / 180)) * b
hata = (b ^ 2 - k ^ 2) ^ 0.5
If b <= TextBox1.Text And b <> 0 Then
If Giris.MSFlexGrid1.TextMatrix(i, 2) = "" Then
If CheckBox3.Value = True Then Giris.MSFlexGrid1.RemoveItem (i + 1)
ListBox1.AddItem (Round(b, 4) & " silindi. Hata = " & (Round(hata, 4)))
If hata > en_buyuk_hata Then en_buyuk_hata = hata
toplam_hatalar = toplam_hatalar + hata
ortalama_hata = toplam_hatalar / ListBox1.ListCount
End If
End If
Else
If b <= TextBox1.Text And b <> 0 Then
If Giris.MSFlexGrid1.TextMatrix(i, 2) = "" Then
If CheckBox3.Value = True Then Giris.MSFlexGrid1.RemoveItem (i + 1)
ListBox1.AddItem (Round(b, 4) & " silindi. Hata = 0")
End If
End If
End If
End If
10
If i = Giris.MSFlexGrid1.Rows - 3 Then Exit For
Next i
ProgressBar1.Value = 0
MsgBox ((ListBox1.ListCount) & " satır silindi")
End Sub
Private Sub CommandButton2_Click()
Me.Hide
End Sub
Private Sub Slider1_Scroll()
TextBox1.Text = Slider1.Value / 100000
End Sub
Private Sub UserForm_Activate()
TextBox1.Text = Slider1.Value / 100000
EndSub

```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : TURGUT, Sevilay
 Uyuğu : T. C.
 Doğum tarihi ve yeri : 01.01.1980, TRABZON
 Medeni Hali : Evli
 Telefon : 0 544 398 32 24
 Fax :
 e-posta : sevilay.turgut@hidromek.com.tr

Eğitim

Derece	Eğitim Birimi	Mevuniyet Tarihi
Lisans	Gazi Üniversitesi / Talaşlı Üretim Öğretmenliği	2004
Lise	Trabzon, Anadolu Meslek Lisesi	1998

İş Deneyimi

Yıl	Yer	Görev
2006 - 2008	Üniversite	Öğretim görevlisi (Sözleşmeli)
2006 - 2008	Mimtek bilgisayar kursları	Eğitimci
2008 -	Hidromek	Mekanik Tasarım Sorumlusu

Yabancı Dil

İngilizce, Almanca

Yayınlar

1. Turgut, S., Turgut, M., “SolidWorks, COSMOSWorks, CosmosMotion, SolidCAM, MoldFlow”, 978-975-02-0977-2, Seçkin Yayıncılık San. Ve Tic. A.Ş., Mayıs-2009

Hobiler

Doğa gezileri, Sosyal etkinlikler, Spor, Resim