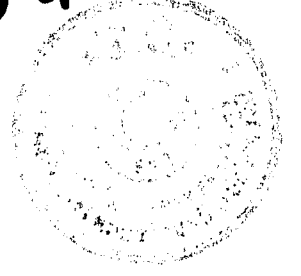


68984



T.C.  
BALIKESİR ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
MAKİNA MÜHENDİSLİĞİ ANABİLİM DALI

BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMIĞI (CFD)  
ANALİZİNİN MÜHENDİSLİK TASARIM AŞAMASI İLE  
BÜTÜNLEŞTİRİLMESİ İÇİN AKTİVİTE VE BİLGİ  
MODELLEMESİ

DOKTORA TEZİ

Yılmaz GÜR

Balıkesir, Mayıs - 1998

T.C.  
BALIKESİR ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
MAKİNA MÜHENDİSLİĞİ ANABİLİM DALI

BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMIĞI (CFD)  
ANALİZİNİN MÜHENDİSLİK TASARIM AŞAMASI İLE  
BÜTÜNLEŞTİRİLMESİ İÇİN AKTİVİTE VE BİLGİ  
MODELLEMESİ

DOKTORA TEZİ

Yılmaz GÜR

Tez Danışmanı : Yrd. Doç. Dr. Ramazan YAMAN

Sınav Tarihi : 29. 05. 1998

Jüri Üyeleri : Prof.Dr.Doğan ÖZGÜR(Yıldız Teknik. Ü.)  
Prof. Sabri SAVAŞ (Balıkesir Üniv.)  
Doç.Dr.Zafer GÜL(Marmara Üniv.)  
Yrd.Doç.Dr.Mehmet İREN(Balıkesir Üniv.)  
Yrd.Doç.Dr. Ramazan YAMAN (Danışman)(BAÜ)  
Yrd.Doç. Dr. Ziya AKSOY (BAÜ)

Balıkesir, Mayıs - 1998



**ÖZ**

**BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMİĞİ (CFD) ANALİZİNİN  
MÜHENDİSLİK TASARIM AŞAMASI İLE BÜTÜNLEŞTİRİLMESİ İÇİN  
AKTİVİTE VE BİLGİ MODELLEMESİ**

**Yılmaz GÜR**

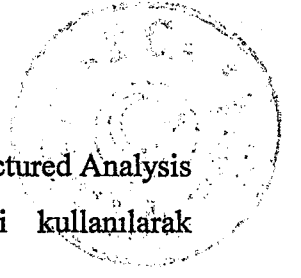
**Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü  
Makina Mühendisliği Anabilim Dalı**

**Ph. D. Tezi / Tez Danışmanı : Yrd. Doç. Dr. Ramazan YAMAN**

**Balıkesir 1998**

Bilgisayar alanındaki, hem donanım hem de yazılım konusunda meydana gelen ilerlemeler, bilgisayar destekli mühendislik analizi gibi mühendislik tasarım prosesinin rutin kısımlarının otomatikleştirilmesi için yeni olanaklar sunmakta ve hatta bu konuda mühendislik bilimi üzerine belirli derecede bir baskı dahi getirmektedir. Mühendislik analiz prosesinin otomatikleştirilmesi ise günümüzde arayüzler oluşturmak suretiyle gerçekleştirilmeye çalışılmakta ise de asıl sorunun çözümü integrasyon (bütünleştirme) kavramında yatmaktadır. İntegrasyonun gerçekleştirilmesi durumunda gereksiz olarak bilginin tekrar edilmesi önlenecek, bilgilerin paylaşımı kolaylaşacak ve yanlış anlaşılmaya meydan verilmeyecektir. İntegrasyon kavramı bizi, analiz için gerekli olacak olan bilgilerin, ürünün yaşam süreci boyunca çeşitli uygulamalar tarafından paylaşılacak olan, merkezi bir bilgi deposunda saklanmasına götürecektir. Fakat böyle bir bilgi kaynağı oluşturulmadan önce, belirli bir uygulamayı gerçekleştirebilmek için gerekli olan aktivitelerin ve bilgilerin araştırılması gerekmektedir.

Bu amaca yönelik olarak, bu çalışma içerisinde CFD analizi ile ilgili aktiviteler ve analizi gerçekleştirebilmek için gerekli olan bilgiler araştırılmıştır. Analiz esnasında



yer alan aktiviteler bir aktivite modelleme tekniđi olan ve SADT (Structured Analysis and Design Technique) diye adlandırılan modelleme yöntemi kullanılarak modellenmiştir.

CFD analizinin gerçekleştirilebilmesi için gerekli olacak bilgiler ise, bir bilgi modelleme yöntemi olan ve EXPRESS-G diye adlandırılan bilgi modelleme metodu yardımıyla modellenmiştir.

İntegrasyon kavramının gerçekleştirilmesinde önemli bir yeri olan Ürün Modelleme ortamı kullanılarak ise bir CFD analizi veri modeli oluşturulmuş ve bu veri modeli ile analizi desteklemek için gerekli olan bilgilerin depolanması sağlanmıştır. Oluşturulan bu CFD analizi veri modeli, bir şofbenin baca gazı çıkışını sağlayan parçasının içerisindeki gaz akışını inceleyen bir problem örnek alınarak test edilmiştir.

**ANAHTAR SÖZCÜKLER :** CFD analizi, Aktivite Modelleme, Bilgi Modelleme, Ürün Modelleme, SADT, Mühendislik Analizi.



## **ABSTRACT**

# **ACTIVITY AND INFORMATION MODELLING OF COMPUTATIONAL FLUID DYNAMICS (CFD) ANALYSIS IN ORDER TO INTEGRATE WITH THE ENGINEERING DESIGN PROCESS**

**Yılmaz GÜR**

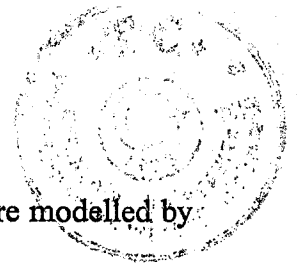
**Balıkesir University, Institute of Science, Department of Mechanical  
Engineering**

**Ph. D. Thesis / Supervisor : Asst. Prof.Dr. Ramazan YAMAN**

**Balıkesir-Turkey, 1998**

Both hardware and software achievements have brought new opportunities and pressures to engineering science to automate the routine part of the engineering design process such as computational engineering analysis. Successful automation of the engineering analysis process is lying under the concept of integration rather than interfacing which is today's current practice. The integration will prevent duplication of information, facilitate the sharing of the information and prevent misunderstanding. The concept of integration will lead us to store the necessary analysis information in a repository such that the information inside of it will be shared by various applications in the product life-cycle. Before such a central source of information is constructed, certain activities and the required information to support a particular application has to be investigated.

Towards this aim, fluid flow analysis activities and information requirements have been investigated within the scope of the thesis. The activities involve with fluid flow analysis have been formalized using the Structured Analysis and Design Technique (SADT).



The required information to realize the fluid flow analysis activities are modelled by using the EXPRESS-G information modelling language.

A CFD analysis data model is proposed to capture the necessary information to support the fluid flow analysis. This is implemented in a Product Model environment and populated using a gas boiler's flue hood example.

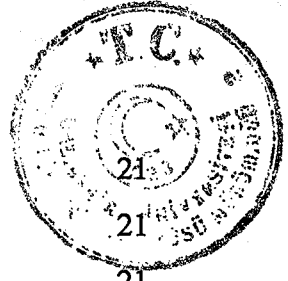
**KEY WORDS :** CFD analysis, Activity Modelling, Information Modelling, Product Modelling, SADT, Engineering Analysis.





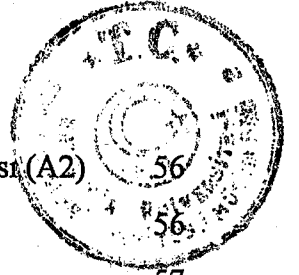
## İÇİNDEKİLER

	<u>Sayfa</u>
ÖZ, ANAHTAR SÖZCÜKLER	ii
ABSTRACT, KEYWORDS	iv
İÇİNDEKİLER	vi
SEMBOL LİSTESİ VE KISALTMALAR	xi
ŞEKİL LİSTESİ	xii
ÖNSÖZ	xiv
ACKNOWLEDGEMENTS	xv
1. GİRİŞ	1
1.1 Çalışmanın Amaçları	1
1.2 Çalışmanın Organizasyonu	2
2. MÜHENDİSLİK TASARIM VE ANALİZ PROSESİ	4
2.1 Giriş	4
2.2 Mühendislik Tasarım Prosesi	4
2.2.1 Mühendislik Tasarımının Tanımları	4
2.2.2 French'e göre Tasarım Prosesi	5
2.2.3 Pahl ve Beitz'a göre Tasarım Prosesi	6
2.2.4 Suh'a göre Tasarım Prosesi	10
2.2.5 Tasarımın Sınıflandırılması	12
2.3 Mühendislik Analiz Prosesi	13
2.3.1 Mühendislik Analizinin Tanımları	14
2.3.2 Mühendislik Analiz Prosesi	15
2.3.3 Mühendislik Analizinin Sınıflandırılması	18
2.4 Bölümsel Sonuç ve Değerlendirmeler	20

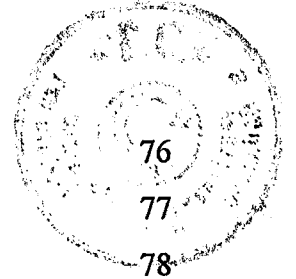


<b>3. ÜRÜN MODELLEME</b>	
3.1 Giriş	
3.2 Bilgisayar Destekli Faaliyetlerin Entegrasyonu	21
3.3 Geometrik Modelleme	22
3.4 Leeds Üniversitesinde Ürün Veri Modellemesi	25
3.4.1 Ürün Veri Model Çerçeve Çalışması	25
3.4.2 Ürün Veri Modelini Hazırlamak İçin Bir Bilgisayar Yazılımı	30
3.4.3 Structure Editor	30
3.5 Ürün Modelleme ile İlgili Çalışmaların Gözden Geçirilmesi	32
3.6 STEP Ürün Modeli Verilerinin Değişimi için Standart	39
3.6.1 SET (Değişim ve Transfer) Standardı	41
3.6.2 VDAFS Standardı	41
3.6.3 PDDI Standardı	42
3.6.4 PDES Standardı	42
3.6.5 ESPRIT Projesi 322 : CAD*I	42
3.6.6 Uluslararası Standartlar Kurumu (ISO)	43
3.6.7 STEP Standardının Metodolojisi ve Amacı	43
3.6.8 STEP Dokümanları	45
3.7 Bölümsel Sonuç ve Değerlendirmeler	48
<b>4. BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMİĞİ (CFD) İÇİN BİR AKTİVİTE ANALİZİ</b>	<b>49</b>
4.1 Giriş	49
4.2 SADT (Structured Analysis and Design Technique)	50
4.3 Akışkan Akışında Yer Alan Aktivitelerinin Analizi	52
4.3.1 Konteks Diyagramı (A-0)	53
4.3.2 Konteks Diyagramının (A0) Ayrıştırılması	54
4.3.3 Fiziksel Modelin Oluşturulması (A1)	54
4.3.3.1 Uygun Fenomenin Seçilmesi (A11)	55
4.3.3.1.1 Aday Fenomenin Belirlenmesi (A111)	55
4.3.3.1.2 İlave Özelliklerin Belirlenmesi (A112)	55
4.3.3.1.3 Ortama Uygunluğun Değerlendirilmesi (A113)	56

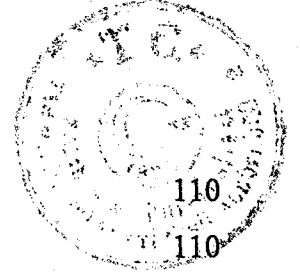




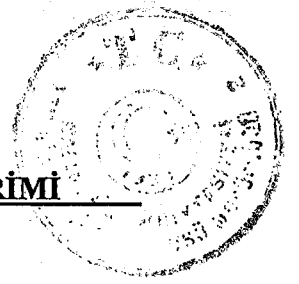
4.3.4 Matematiksel Modelin Oluşturulması Aktivitesinin Ayrıştırılması (A2)	56
4.3.4.1 CFD Problem Bilgilerinin İdealleştirilmesi (A21)	
4.3.4.1.1 Boyutların İdealleştirilmesi (A211)	57
4.3.4.1.2 Akışkan Özelliklerinin İdealleştirilmesi (A212)	57
4.3.4.1.3 CFD Alan Geometrisinin İdealleştirilmesi (A213)	57
4.3.4.1.4 Sınır ve Başlangıç Koşullarının İdealleştirilmesi (A214)	58
4.3.4.2 Denklemlerin Formülasyonu (A22)	58
4.3.4.2.1 Temel Denklemlerin Çıkarılması (A221)	58
4.3.4.2.2 Sistem Denklemlerinin Boyutsuz Hale Getirilmesi (A222)	59
4.3.4.2.3 Çözüm Teknikleri ve Stratejilerinin Seçimi (A223)	60
4.3.5 Analiz Aşaması (A3)	60
4.3.5.1 Matematiksel Modelin Diskretizasyonu (A31)	61
4.3.5.2 Denklemlerin Çözülmesi (A32)	61
4.3.5.3 Elemanlara Ayırma Hatalarının Kontrol Edilmesi (A33)	62
4.3.6 Sonuçların Bilgisayar Ortamında İşlenmesi (A4)	62
4.3.7 İdealleştirme Hatalarının Kontrolü (A5)	62
4.4 Bölümsel Sonuç ve Değerlendirmeler	63
5.BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMİĞİ (CFD)	
ANALİZİ İÇİN BİR BİLGİ MODELİ	72
5.1 Giriş	72
5.2 Bilgi Modellemesi	72
5.2.1 Bilgi Modellemenin Faydaları	73
5.2.2 Bilginin Temsil Edilmesi	74
5.2.3 Nesnelere Grafiksel olarak veya Kelimelerle ifade etmenin avantaj ve dezavantajları	74
5.3 CFD Analizi İçin Bir Bilgi Modeli	75
5.3.1 Fiziksel Model	75
5.3.1.1 Fiziksel Fenomen	75
5.3.1.2 Fiziksel Teoriler	75
5.3.1.3 Kabuller	76
5.3.1.4 Mühendislik Yargısı	76



5.3.1.5 Prensipler ve Sınırlamalar	76
5.3.2 Matematiksel Model	77
5.3.2.1 Akışkan Özellikleri	78
5.3.2.2 CFD Sınır ve Başlangıç Koşulları	79
5.3.2.3 CFD Alan Geometrisi	79
5.3.2.4 CFD Alan Geometrisinin Tanımlanması	80
5.3.2.5 Matematiksel Denklemler	80
5.3.2.6 Çözüm Teknikleri	81
5.3.2.7 Çözüm Stratejileri	81
5.4 Bölümsel Sonuç ve Değerlendirmeler	81
6. CFD ANALİZİ VERİ MODELİNİN ÜRÜN MODELLEME ORTAMINDA OLUŞTURULMASI	92
6.1 Giriş	92
6.2 Yazılımın Hazırlanmasında Kullanılan Bilgisayar Ortamı	93
6.3 CFD Analiz Veri Modelinin Şematik Yapısı	93
6.4 CFD Veri Modelinin Ürün Modelleme Ortamına Entegrasyonu İçin Arayüz Oluşturulması	96
6.5 CFD Analizi Veri Modelinin Örneklenmesi	97
6.5.1 Girdi Dosyasındaki Bilgilerin Tanımlanması	98
6.6 Bölümsel Sonuç ve Değerlendirmeler	103
7. SONUÇ VE ÖNERİLER	104
7.1 Giriş	104
7.2 Çalışmanın Sağladığı Katkıları	104
7.3 İleride Yapılacak Çalışmalar İçin Öneriler	105
EKLER :	
EK-A: SADT AKTİVİTE MODELLEME TEKNİĞİ	107
A.1 Diyagram Numaraları	108
A.2 ICOM Notasyonu	109



<b>EK B: EXPRESS-G BİLGİ MODELLEME DİLİ</b>	
B.1 Tanımlama Sembolleri	110
B.1.1 EXPRESS'teki basit tip semboller	110
B.1.2 Tip Sembolleri	111
B.1.3 Varlık Sembolü	111
B.1.4 Şema Sembolü	112
B.1.5 İlişki Sembolleri	112
B.2 Sayfa Düzenleme Sembolleri	113
B.2.1 Bir Sayfayı Referans Verme	113
<b>EK C: Bir Gaz Şofbenine Ait Baca Parçasının Fidap CFD Analiz Programı İle Analizi İçin Hazırlanan Girdi Dosyası</b>	115
<b>EK D: CFD Analiz Verilerini Ürün Modelleme Ortamından Manipüle Etmek İçin Kullanılan Yazılımdan Örnek Teşkil Etmesi Amacıyla Alınmış Program Kesiti</b>	118
<b>EK E: Oluşturulan CFD Analizi Veri Modelinin Örnek Bir Uygulama İçin Çalıştırılması Sonucu Elde Edilen Çıktılar</b>	129
<b>KAYNAKÇA</b>	138

**SEMBOL LİSTESİ****TANIMI****BİRİMİ**

$\mu$	Mutlak viskozite	kg/m.s
$\nu$	Kinematik viskozite	m <sup>2</sup> /s
$\rho$	Yoğunluk	kg/m <sup>3</sup>
$p$	Basınç	kg/m.s <sup>2</sup>
Re	Reynolds Sayısı	---

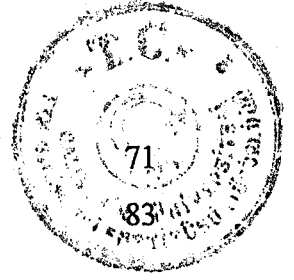
**KISALTMALAR****TANIMI**

AP	Uygulama Protokolleri (Application Protocol)
B-Rep	Sınırları İle Bir Nesnenin İfade Edilmesi (Boundary Representation)
CFD	Bilgisayar Destekli Akışkanlar Dinamiği (Computational Fluid Dynamics)
CSG	Katı Cisim Geometrisi Oluşturulması (Constructive Solid Geometry)
FIDAP	Akışkanlar Dinamiği Analiz Paketi (Fluid Dynamics Analysis Package)
IDEF	Tümleşik Bilgisayar Destekli İmalatı Tanımlama Metodu (Integrated Computer Aided Manufacturing Definition Method)
IGES	İlk Grafik Değişim Spesifikasyonu (Initial Graphics Exchange Specification)
PDES	Ürün Veri Değişim Standardı (Product Data Exchange Standard)
SADT	Yapısal Analiz ve Tasarım Tekniği (Structured Analysis and Design Technique)
SE	Veri Editörü (Structure Editor)
STEP	Ürün Verilerinin Değişimi Standardı (Standard for the Exchange of Product Model Data)

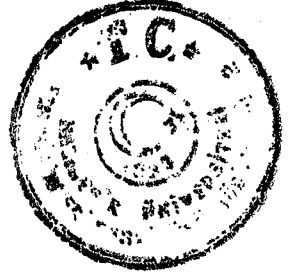
## ŞEKİL LİSTESİ



Şekil Numarası	Adı	Sayfa
Şekil 2.1	French'e göre tasarım prosesinin aşamaları [1]	7
Şekil 2.2	Pahl ve Beitz'a göre tasarım prosesinin aşamaları [2]	9
Şekil 2.3	Fonksiyonel uzaydan fiziksel uzaya gömme prosesi [3]	11
Şekil 2.4	Suh'a göre dizayn prosesi [3]	11
Şekil 2.5	Tworzydlo'ya göre mühendislik analiz prosesi [5]	17
Şekil 3.1	Bilgisayar destekli aktivitelerin integrasyon trendi [17]	22
Şekil 3.2	Bir nesnenin CSG modelleme yöntemi ile ifadesi	24
Şekil 3.3	Bir nesnenin B-Rep modelleme yöntemi ile ifadesi	24
Şekil 3.4	Ürün veri modeli mimarisi	26
Şekil 3.5	Ürün veri modeli çerçeve çalışması	28
Şekil 3.6	Ürün veri modelindeki düzeyler	29
Şekil 3.7	Ürün veri modelindeki tekrar eden kalıplar	30
Şekil 3.8	Spur'ün ürün modeli [47]	32
Şekil 3.9	Makina imalatı için ürün modeli [44]	33
Şekil 3.10	Kimura'nın ürün modelinin sistem mimarisi [51]	34
Şekil 3.11	A.S.U.test yatağının sistem mimarisi [46]	35
Şekil 3.12	CAD*I ürün modelinin mimarisi [48]	37
Şekil 3.13	Gu'nun genel ürün modelinin mimarisi [49]	38
Şekil 3.14	Veri değişim formatının evrimi [53]	40
Şekil 3.15	STEP dokümantasyonunun mimarisi [53]	46
Şekil 4.1	SADT diyagramlarının hiyerarşik yapısı	51
Şekil 4.2	CFD Analizi (Konteks) ; Endeks No A-0	64
Şekil 4.3	CFD Analizi; Endeks No A0	65
Şekil 4.4	Fiziksel modelin oluşturulması; Endeks No A1	66
Şekil 4.5	Uygun fenomenin seçilmesi; Endeks No A11	67
Şekil 4.6	Matematiksel modelin oluşturulması; Endeks No A2	68
Şekil 4.7	CFD problem bilgilerinin idealleştirilmesi; Endeks No A21	69
Şekil 4.8	Denklemleri formüle et ; Endeks No A22	70



Şekil 4.9	CFD analiz aşaması; Endeks No A3	
Şekil 5.1	CFD analizi için kısmi bir bilgi modeli diyagramı	
Şekil 5.2	Akışkan özellikleri ve CFD alan geometrisi için kısmi bilgi modeli	84
Şekil 5.3	Geometri tanımlama ile ilgili kısmi bilgi modeli	85
Şekil 5.4	Akışkan özellikleri ile ilgili kısmi bilgi modeli	86
Şekil 5.5	Matematiksel denklemlerle ilgili kısmi bilgi modeli	87
Şekil 5.6	Çözüm stratejileri için kısmi bilgi modeli	88
Şekil 5.7	Ayrılmış (Segregated) çözüm stratejileri için kısmi bilgi modeli	89
Şekil 5.8	Sınır ve başlangıç koşulları için kısmi bilgi modeli	90
Şekil 5.9	Kontrol bilgileri için kısmi bilgi modeli	91
Şekil 6.1	CFD veri modelinin şematik yapısı	94
Şekil 6.2	Analizde kullanılan parçanın kesiti	97
Şekil 6.3	2 Boyuta indirgenmiş örnek problemin sonlu elemanlara ayrılmış biçimi	100
Şekil 6.4	Analiz edilen parça içerisindeki gaz akışı	102
Şekil A.1	Bir SADT kutusu ve okları	108
Şekil B.1	Basit tip semboller	111
Şekil B.2	Tip tanımlama sembolleri	111
Şekil B.3	Varlık tanımlama sembolü	112
Şekil B.4	Şema tanımlama sembolü	112
Şekil B.5	İlişkileri ifade etmede kullanılan çizgi tipleri	112
Şekil B.6	Ağaç diyagramındaki ilişkiler	113
Şekil B.7	EXPRESS-G bilgi modelleme tekniğinde bir sayfayı referans verme	114
Şekil B.8	EXPRESS-G ile oluşturulan CFD bilgi modelinde bir başka sayfayı referans gösterme	114



## ÖNSÖZ

Öncelikle, lisans düzeyinde de öğrencisi olduğum, ve aynı zamanda bir süre doktora danışmanlığımı da üstlenen değerli hocam Prof. Sabri SAVAŞ'a ve ikinci danışmanım sayın hocam Yrd. Doç. Dr. Ramazan YAMAN'a çalışmam boyunca bana sağlamış oldukları destek, yardım ve yönlendirmelerinden dolayı teşekkür ederim.

Ayrıca sağladıkları destek nedeniyle Leeds Üniversitesi Makina Mühendisliği bölümündeki Bilgisayar Destekli Mühendislik araştırma grubu elemanlarına da teşekkürlerimi sunarım.

Çalışmam süresince maddi ve manevi desteğini eksik etmeyen eşim Hülya ve kızım Begüm'e, ayrıca hayatım süresince bana her türlü desteği sunan ve bana cesaret veren anne ve babama da sevgilerimi sunarım.

Balıkesir, 1998

Yılmaz GÜR



## ACKNOWLEDGEMENTS

First of all, I would like to thank to my respectable professor Prof. Sabri SAVAŞ, who was one of my professors in my undergraduate study, and he also supervised my Ph.D. study for a certain period of time, and my second supervisor Asst. Prof. Dr. Ramazan YAMAN for their support, and help throughout the preparation of this thesis.

I would like to thank all the members of the CAE research group in the Department of Mechanical Engineering of Leeds University, for their kind help and guidance.

Finally, I would like to thank my wife, Hülya, for her love, support and patience over the past years, and my amazing daughter, Begüm. My parents also deserve my special thanks for their continuous support and encouragement.

Balıkesir, 1998

Yılmaz GÜR





## 1. GİRİŞ

Mühendislikte tasarımının başarısı, tasarımın davranışını önceden tahmin etmeye bağlıdır. Bir mühendislik tasarımının davranışını önceden tahmin etmenin yollarından biri de bu tasarımın bilgisayar ortamında benzetimini yapmaktır. Son yıllarda, bilgisayar teknolojisindeki ve bilgisayara dayalı çözüm metodlarında kaydedilen ilerlemeler, sayısal modellerin oluşturulmasını ve fiziksel bir problemin oldukça kolay bir şekilde benzetiminin yapılmasını mümkün hale getirmiştir. Bilgisayar ortamındaki bu tür bir benzetimin, mühendisin, bir fiziksel sistemin davranışını farklı koşullar altında, nispeten kısa bir zaman dilimi içinde ve daha ucuz bir şekilde, tahmin etmesini sağlar. Fakat günümüzde kullanılmakta olan mühendislik analiz yazılım programlarının kötü yanı, her yazılımın kendine özgü bir veri ortamına ihtiyaç duymasındır. İşte bu veri ortamlarının bütünleştirilmesi ise bilgisayar destekli mühendislik tasarım prosesinin otomatikleştirilmesini sağlamak üzere yol gösterebilir.

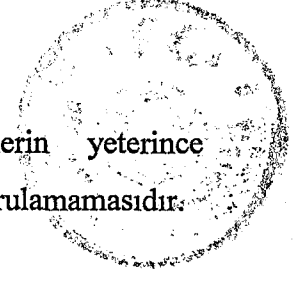
Bütünleştirme aynı zamanda bir ürünün piyasaya çıkış süresini ve hızla değişen pazar isteklerine cevap verme zamanını kısaltacaktır. Ürün modelleme (Product Modelling) olgusu bu entegrasyon problemini çözebilecek bir kavramdır.

Bu kavram, çalışmanın üçüncü bölümünde detaylı olarak incelenmiş ve ürün modelleme ortamında yapılan CFD analizi veri modeli altıncı bölümde açıklanmıştır.

### 1.1 Çalışmanın Amaçları

Şu anda, bilgisayar destekli akışkanlar dinamiği analizi (Computational Fluid Dynamics) mühendislik tasarımı aşamasına bazı eksikliklerden dolayı integre

edilememektedir. Bu eksiklikler, gerekli bilgi ve aktivitelerin yeterince tanımlanamaması ve ürün modelleme ortamının istenen düzeyde oluşturulamamasıdır.



Bu amaca yönelik olarak, Ürün Modelleme olgusu, problemin merkezi bir bilgi kaynağı tarafının çözümlenmesine, ve analizdeki aktivitelerin ve bilgilerin modellenmesi ise analizi destekleyecek gerekli bilgilerin ve aktivitelerin daha iyi tanımlanıp anlaşılmasına yardım edecektir.

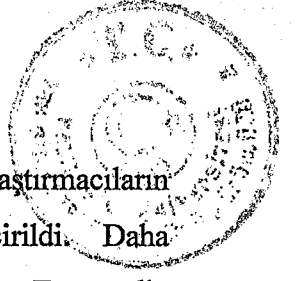
Bu çalışma aşağıdaki amaçlara yöneliktir;

- CFD analizi aktivitelerini tanımlayıp onların daha iyi anlaşılmasını sağlamak üzere SADT (structured analysis and design technique-yapısal analiz ve tasarım tekniği) tekniğini kullanarak bir aktivite modeli hazırlamak,
- CFD analizinin yapılabilmesi için gerekli olabilecek bilgilerin tanımlandığı bir bilgi modeli (information model) oluşturmak,
- Analizin yapılmasını sağlamak üzere Ürün Modelleme ortamında Structure Editor denilen yazılımı kullanarak bir CFD analizi veri modeli geliştirmek.

Yukarıda maddeler halinde verilen katkıların yerine getirilmesi ile bilgisayar destekli akışkanlar dinamiği (CFD) analizinin mühendislik tasarımına bütünleştirilmesi için bir yaklaşım geliştirilecektir.

## 1.2 Çalışmanın Organizasyonu

Bu çalışmada, bilgisayar destekli akışkanlar dinamiği (CFD) analizi için bir aktivite ve bilgi modeli sunulurken aynı zamanda bütünleştirme uygulamasına yönelik bir CFD analizi veri modeli geliştirildi.



Bölüm 2’de, öncelikle French [1], Pahl&Beitz [2] ve Suh [3] gibi arařtırmacıların alıřmaları yardımıyla mühendislik tasarım ve analiz prosesi gözden geçirildi. Daha sonra, mühendislik analiz prosenin adımları Shepard ve arkadaşları [4] ve Tworzydło ve arkadaşları [5]’nun alıřmalarından faydalanılarak belirlendi.

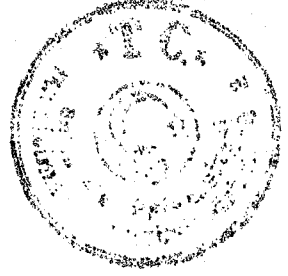
Bölüm 3’de, bir ürünün tasarım aşamasından, ürünün kullanımdan çıkarıldığı ana kadar geen süre içerisinde yapılabilecek olan tüm mühendislik uygulamalarını destekleyebilecek biçimde gerekli bütün bilgileri içerebilecek Ürün Modelleme olgusu incelenmektedir. Bu bölümde aynı zamanda, mühendislik uygulamaları arasında halen kullanılmakta olan veri alış-veriři için belirlenmiş veri alış-veriři standartları da gözden geçirilmiştir.

Bölüm 4’de ise, SADT tekniđi kullanılarak oluşturulan CFD analizi için aktivite modeli detaylı olarak verilmiş ve sözü edilen bu teknik sayesinde, bazı biçimsel kurallara riayet edilerek, aktivitelerin istenilen derinlikteki detaya ulaşana kadar daha alt seviyelerdeki alt aktivitelere ayrılması ve bunların grafik olarak izah edilmesi mümkün olmuştur.

Bölüm 5’de, EXPRESS-G denilen bir grafiksel bilgi modelleme dili kullanılarak yapılan akışkanlar dinamiđi analizinde gerekli olan bilgilerin modellendiđi bir modelin hem grafik olarak hem de metin olarak izahı yapılmıştır.

Bölüm 6’da ise, bilgisayar destekli akışkanlar dinamiđi (CFD) analizi için gerekli olabilecek bilgilerin depolanabileceđi bir veri tabanının grafik ortamda hazırlanmasına imkan sađlayan Structure Editor denilen bir veri tabanı editorü kullanılarak, bir veri tabanı oluşturuldu. Bir sonraki aşamada da, bu yaklaşımı örnekleme amacıyla bir problem FIDAP olarak adlandırılan bir bilgisayar ile akışkanlar dinamiđi analizi yazılım paketi kullanılarak test edildi.

Son olarak, yedinci bölümde de yapılan alıřma kısaca özetlenirken, tezdeki alıřma ile sađlanan katkılar ve ileride bu konuda yapılabilecek arařtırmalar hakkında bazı tavsiyelerde bulunuldu.



## 2. MÜHENDİSLİK TASARIM VE ANALİZ PROSESİ

### 2.1 Giriş

Tasarım, mühendislikte merkezi bir faaliyettir [6]. Fakat hiç şüphesiz ki tasarım karmaşık bir kavramdır ve bu tasarımın anlaşılmasını daha da güçleştirmektedir [7]. Bütün tasarım işlemlerinin asıl amacı, tasarım özelliklerini en iyi şekilde karşılayabilecek bir çözüme ulaşmaktır.

İzleyen kısımda, mühendislik tasarımının ne olduğu tanımlanacak ve French [1], Pahl & Beitz [2] ve Suh [3]'a göre tasarım prosesi gözden geçirilecektir. Takibinde ise mühendislik analizi prosesi tanımlanacak, mühendislik tasarımı işleminde yer alan hali hazırdaki mevcut mühendislik analizi vurgulanacak ve Shepard [4], Tworzydło [5]'nin bu konuda yaptığı çalışmalara dayalı olarak mühendislik analizi prosesi incelenecektir.

### 2.2 Mühendislik Tasarım Prosesi

#### 2.2.1 Mühendislik Tasarımının Tanımları

Aşağıdaki tanımlardan da görüleceği üzere her araştırmacı tasarımı, onun kompleks doğası nedeniyle, kendine özgü bir biçimde tanımladı.

Dym and Levitt [8] mühendislik tasarımını;



“Belirlenen amaçlara ulaşmak ve verilen şartları sağlamak üzere insan tarafından üretilen şeylerin biçim ve fonksiyonlarını belirleyen özelliklerin değerlendirilmesi ve zekaya dayalı üretimi için bir sistematik” olarak tanımlamaktadır.

French [1] ise onu;

“Bir mühendislik ürününün formunu tanımlayan detayların belirtilmesi, tasviye, sıralama, hesaplama, simulasyon, keşif, algılama işlemlerinin tümü” olarak tanımlamaktadır.

Woodson [9] tarafından ise;

“İnsan ihtiyaçlarını karşılamak üzere, tercihan kaynakların optimum sistemlere veya aletlere dönüştürüldüğü planları üretecek iteratif bir karar verme faaliyeti” olarak tanımlanmıştır.

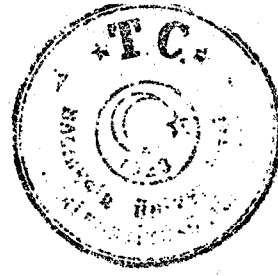
Her ne kadar yukarıdaki bu tanımlar benzer anlamlara sahip olsalar da konunun tanımlanması için standart bir tanım ise bulunmamaktadır.

### 2.2.2 French'e göre Tasarım Prosesi

French [1], tasarım işleminin en çok referans verilen modellerinden birini ortaya koymuştur. Bu model Şekil 2.1 'de gösterilmiştir. Bu modelde daireler aşamaları, dikdörtgenler de faaliyetleri temsil etmektedir. French tasarım işlemini dört sayfaya ayırır;

- **Problemin analizi safhası**

Bu bölümün çıktısı problemin tanımlanmasıdır. Problemin ifade edilmesi üç şekilde olabilir:



1. Problemin tam olarak ifade edilmesi.
2. Çözüm üzerine konulan sınırlamalar.
3. Üzerinde çalışılacak mükemmellik kriteri.

- **Kavramsal Tasarım**

Bu aşamada tanımlanan problem alınır ve ona çok genel anlamda kavramsal seviyede çözümler üretilir. Bu aşama mühendislik biliminin, pratik bilginin, üretim metodlarının kullanıldığı ve çok önemli kararların alındığı bir aşamadır.

- **Planların Somutlaştırılması**

Bu aşamada ise kavramlar daha detaylı olarak çalışılır, incelenir ve eğer birden fazla uygun plan varsa bunlar arasında bir seçim yapılır. Son ürün ise genellikle detay çizimlerdir. Bu aşamadan kavramsal tasarım aşamasına bir geri besleme vardır.

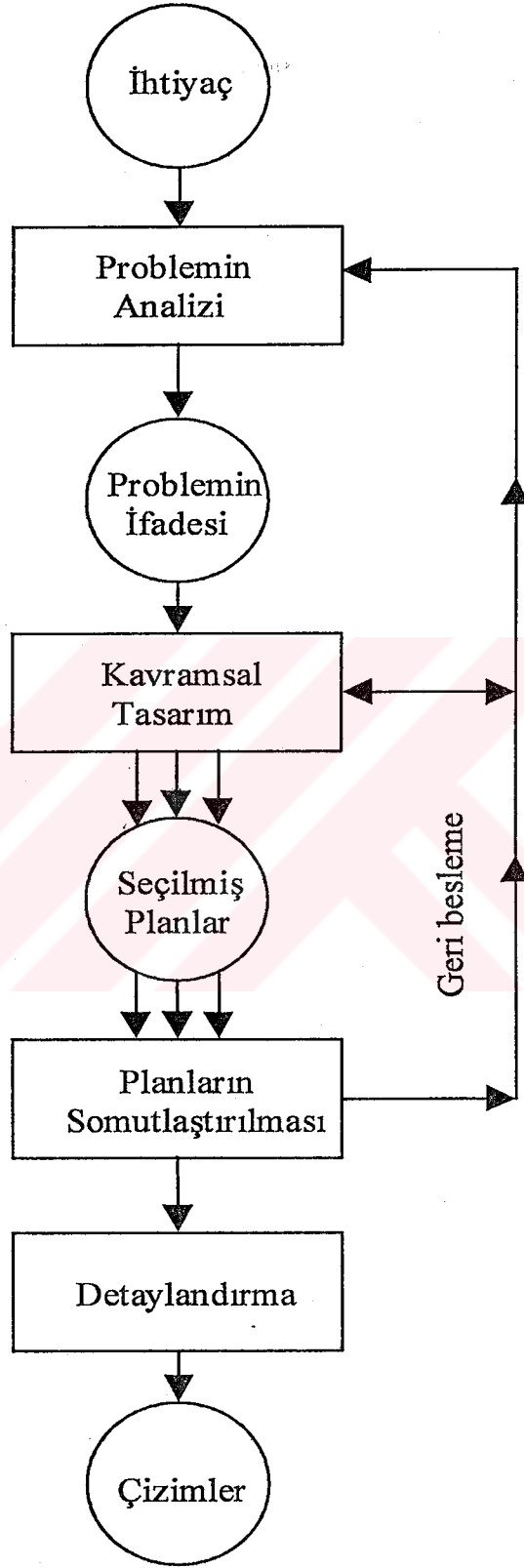
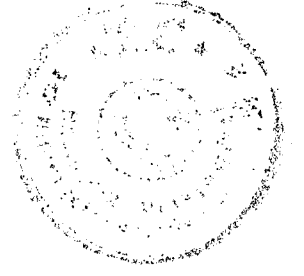
- **Detaylandırma**

Detaylandırma aşamasında, ilgili bilginin çoğu formüller, el kitapları, algoritmalar, veri tabanları, kataloglar gibi özel kurallarla ifade edilirler.

### **2.2.3 Pahl & Beitz'a göre Tasarım Prosesi**

G. Pahl ve W. Beitz [2], sistematik prosedüleri ve gelişmekte olan tasarım kurallarını uygulamadaki deneyimlerini kullanarak mühendislik tasarımı için bir strateji hazırlamışlardır. Burada tasarım prosesi dört farklı ana kategoriye ayrılmaktadır. Bunlar sırasıyla;

1. yapılacak işin netleştirilmesi,
2. kavramsal tasarım,



Şekil 2.1 French'e göre tasarım prosesinin aşamaları [1]



3. somutlaştırılmış tasarım,
4. detaylı tasarımdır.

Bu dört aşama ve bu aşamalar içinde yer alan faaliyetler Şekil 2.2 de gösterilmiş ve aşağıda daha detaylı olarak açıklanmıştır.

#### • Yapılacak İşin Netleştirilmesi

Yapılacak işin netleştirilmesi bölümü, çözüme dahil edilecek kısıtlamalar ve gereksinimler hakkındaki bilgilerin açığa çıkarılmasını ve özelliklerin detaylı olarak hazırlanmasını kapsamaktadır. Spesifikasyonlar (veya gereksinimler) talepler veya istekler olarak tanımlanmalıdır. Talepler gerçekleşmesi zorunlu ve taviz verilemez olan şeylerdir. Bütün koşullar altında mutlaka karşılanmalıdırlar. Talepleri karşılayamayan çözümler ise kabul edilemezler. İstekler ise mümkün olması durumunda dikkate alınmalıdırlar. Ayrıca isteklerin önem sırası da sınıflandırılmalıdır.

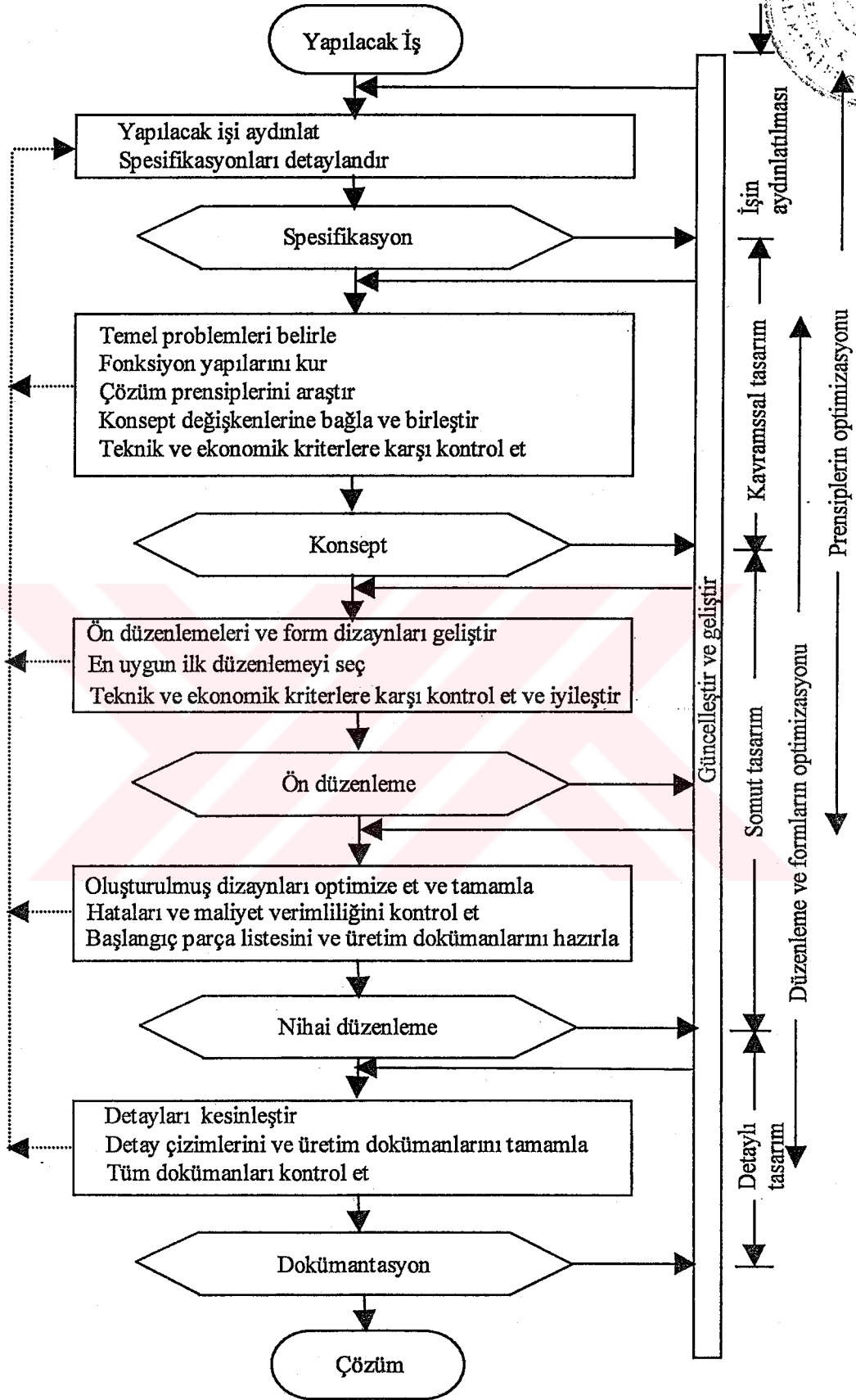
#### • Kavramsal Tasarım

Tasarım işleminin bu aşamasında, fonksiyon yapıları, çözüm prensiplerinin araştırılması ve onların kavram değişkenleri ile birleştirilmesi işlemleri yerine getirilir.

Kavramsal tasarım için aşağıdaki adımlar önerilmektedir;

1. esas problemin tanımlanabilmesi için özet çıkarma,
2. fonksiyon yapısını ortaya koyma,
3. çözüm prensiplerinin araştırılması,
4. çözüm prensiplerinin kombinasyonu,
5. uygun kombinasyonların seçilmesi,
6. kesin bir biçimde kavram değişkenlerine dönüştürülmesi,
7. kavram değişkenlerinin değerlendirilmesi.





Şekil 2.2 Pahl ve Beitz'a göre tasarım prosesinin aşamaları [2]



Bu aşamadan çıkan ise bir kavramdır.

- **Somut Tasarım**

Bu aşamada, kesin plan, ilk tasarım biçimi (parça şekilleri ve malzemeler) belirlenir ve teknik, ekonomik hususlar göz önünde tutularak teknik bir ürün geliştirilir. Genellikle, teknik, ekonomik değerlendirmeler ve değişik alternatiflerin avantaj ve dezavantajları hakkında daha fazla bilgi edinmek için birden fazla planın belirlenmesi gereklidir.

Somut tasarım aşaması, analiz ve sentez adımlarının, sayesinde birbirlerini değiştirttikleri ve tamamladıkları çok sayıdaki geri beslemeler içermektedir.

- **Detaylı Tasarım**

Bu aşama, düzenleme, form, boyut, malzeme, yüzey özelliklerinin belirlendiği ve fizibilitenin teknik ve ekonomik kriterlere göre yeniden kontrol edildiği tasarımın son aşamasıdır. Ayrıca, tüm çizimler ve diğer üretim dokümanları da bu aşamada yapılır.

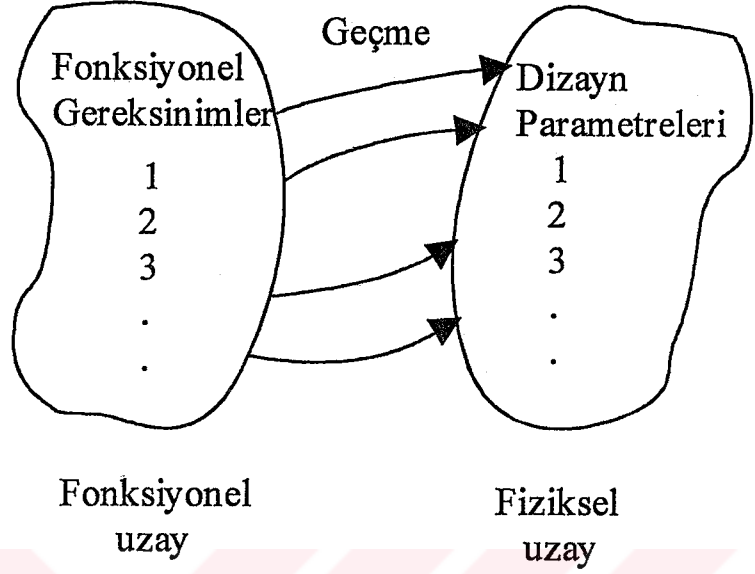
#### 2.2.4 Suh'a göre Tasarım Prosesi

Suh [3], fiziksel çözümün fiziksel uzayda gerçekleştiğini ve bir dizi tasarım parametresi (DP's = Design Parameters) tarafından belirlendiğini, tasarımın amacının, fonksiyonel bir uzayda bir dizi fonksiyonel gereksinim (FR's = Functional Requirements) tarafından betimlendiğini öne sürmektedir. Tasarım süreci ise, doğal olarak birbirinden, tasarım faaliyetinin her hiyerarşik seviyesinde, bağımsız olan bu iki uzay arasındaki ilişkisel bağlantı ile uğraşmaktadır (Şekil 2.3).

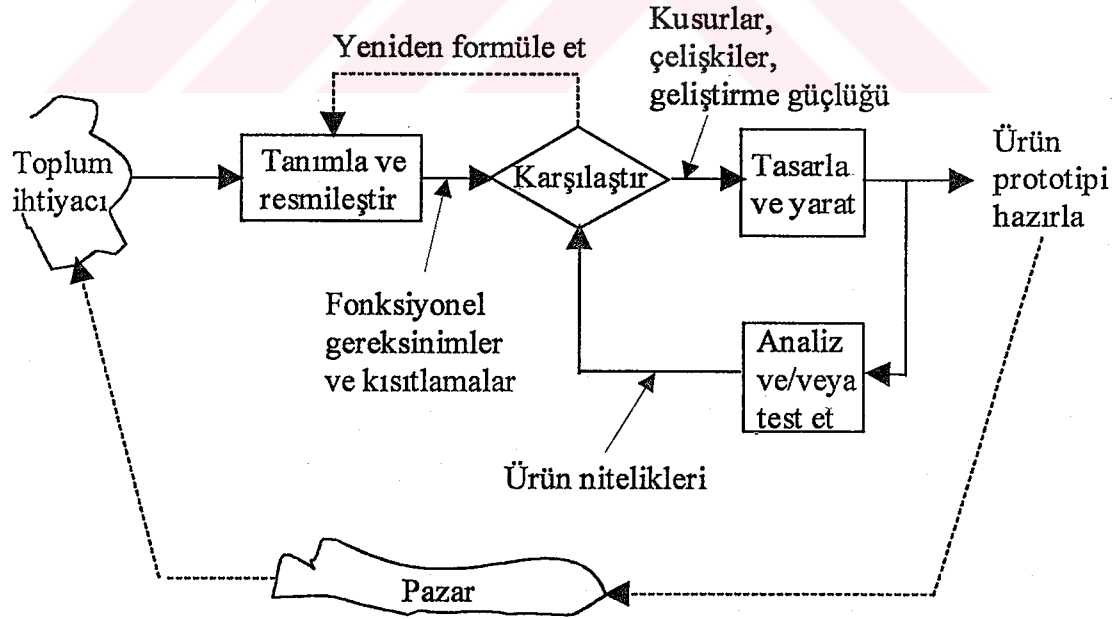
Şekil 2.4 den de görüleceği üzere, Suh'un iteratif tasarım süreci sosyal ihtiyaçların tanımlanmasıyla başlar ve daha sonra bu ihtiyaçların fonksiyonel gereksinimler olarak formülize edilmesi ile devam eder. Planların yapılması ve resmileştirilmesi tamamlandıktan sonra ürün analiz edilir ve bir geribesleme çevrimi ile orijinal işlevsel



ihtiyaçlarla karşılaştırılır. Bu iteratif proses, işlevsel gereksinimler yaratılan ürün tarafından tam olarak sağlanıncaya kadar devam eder.



Şekil 2.3 Fonksiyonel uzaydan fiziksel uzaya gömme prosesi [3]



Şekil 2.4 Suh'a göre dizayn prosesi [3]



Ancak, Suh'un tasarım prosesi, prosesin başında yapılan iteratif analiz işlemlerinin ileriki sahalarda yapılanlardan nasıl ayrıldığı konusunda bir bilgi vermemektedir. Onun tasarım prosesinde, ürün bilgisi analiz faaliyetine girdi olarak girmektedir. Fakat, tasarımın ilk aşamalarındaki ürün bilgisi ile daha sonraki aşamalarda kullanılan ürün bilgisi birbirinden farklılıklar göstermektedir.

### 2.2.5 Tasarımın Sınıflandırılması

Günümüzde yapılan tasarımları bazı tiplere ayırarak onları sınıflandırmak mümkün olabilmektedir. Örneğin Dym [6] mühendislik tasarımını aşağıdaki gibi sınıflandırmaktadır;

- **Yaratıcı (Sınıf 1) Tasarım**

Yaratıcı tasarım yeni çözümler ve yeni ürünler bulma anlamını taşımaktadır. Bu sınıf bir tasarım, tipik olarak hem o alandaki bilgi eksikliği hem de problem çözüm strateji bilgisindeki eksikliğiyle karakterize edilir.

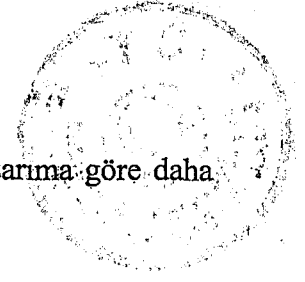
- **Varyant (Sınıf 2) Tasarım**

Varyant tasarım ise, varolan bir tasarımın yeniden düzenlenmesi ve/veya boyutlarının değiştirilmesini kapsamaktadır. Bu sınıf tasarımda, tasarımcı tasarım alanında oldukça iyi bir bilgiye sahiptir ve tasarım bilgisinin kaynaklarının da farkındadır. Ancak sahip olduğu bu bilgilerin gündemdeki problem için nasıl kullanılacağı konusunda eksikliği vardır.

- **Rutin (Sınıf 3) Tasarım**

Alışıl gelmiş tasarım diye de adlandırabileceğimiz bu tasarım sınıfında ise, bilinen çözüm veya çözümler farklı bir ihtiyaca cevap verecek şekilde geliştirilirler. Bu tip tasarımda, bir tasarımı tamamlamak için gerekli olan, (örneğin tasarım bilgisi ve o

bilginin nasıl uygulanacağı gibi), herşey bilinmektedir ve yaratıcı tasarıma göre daha kolay gerçekleştirilebilir.



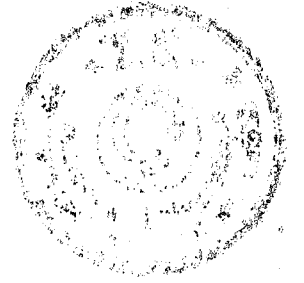
Pahl & Beitz [2], makina mühendisliği endüstrisindeki tasarımların % 55'inin rutin (yani adaptif) tasarım, % 25'inin yeni yaratılmış yani orijinal tasarım, % 20'sinin ise varyant (yani varolan bir tasarımın yeniden ayarlanmış) tasarım tipinde olduğunu söylemektedir. Yani, bugün yapılan tasarımların % 75'i varolan tasarımların tekrarından başka bir şey değildir.

### 2.3 Mühendislik Analiz Prosesi

Analiz mühendislik tasarım prosesinin önemli bir bölümüdür. Analiz işlemi olmadan yapılacak olan tasarımın gerçek hayatta nasıl davranacağına yönelik herhangi bir çalışma ise sadece duyumsal kararlara ve tahminlere dayalı olacaktır. Mühendislik analiz işlemi, bir yandan tasarımın nasıl davranacağı konusunda bize somut bilgiler verirken diğer yandan da tasarımcıya analiz kararlarını nasıl değiştirebileceği konusunda rehberlik edebilecektir[10].

Günümüzde, bilgisayarların çok hızlı bir şekilde artan gücü sayesinde bilgisayar destekli nümerik analiz ve bilgisayar ortamında simulasyon kavramları kolayca telaffuz edilebilir hale gelmiştir. Bu yüzden günümüzde, bilgisayar destekli mühendislik analizi kavramı, tasarım prosesinde şimdiye kadar olmadığı biçimde, önemli bir rol oynamaya başlamıştır.

Bu bölümün amacı ise, mühendislik tasarım prosesinin içinde, mühendislik analiz aşaması olarak adlandırılacak bir faaliyet alanının bulunduğunu vurgulamaktır.



### 2.3.1 Mühendislik Analizinin Tanımları

Genel olarak mühendislik analizi Oxford Sözlüğünde [11] ;

"bir şeyin ona ait parçalarını ve bunlar arasındaki ilişkilerin incelenmesi ve bunun sonuçlarının ifade edilmesi için yapılan bir çalışma"

olarak ifade edilmektedir.

Dym [6] ise bu tanımı şöyle yapmaktadır;

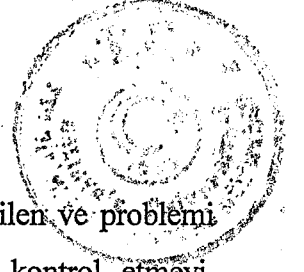
"Analiz, bir ilk veya somut tasarımın veya güncel bir sentezin davranışlarını değerlendirmek üzere ihtiyaç duyulan hesaplamaların yapıldığı bir işdir."

Brooke [12] mühendislik bağlamı içinde şöyle bir analiz tanımı sunmaktadır:

"Bir mühendislik sisteminin elemanlarının davranışını tahmin etmek üzere, fiziksel bir olayın matematik modelinin kullanılması."

Fakat Brooke'un bu tanımı, resmi olmayan gündelik kullanımda analiz diye adlandırılan ve bunun için yapılan bir takım aritmetik işlemi kapsamamaktadır. Brooke aynı zamanda bir nesnenin yoğunluğunu kullanarak kütlelerini hesaplamasının bir analiz olmadığını bunun daha doğru bir deyimle sadece bir *hesaplama* olduğunu belirtmektedir.

Tasarımla ilgili literatürde, *Analiz* mühendislik tasarım işleminin bir yönü olarak algılanmaktadır. Simon [13] çalışmasında mühendislik tasarım prosesinde *Analiz ve Test* diye adlandırdığı özel bir aşamaya yer vermektedir. Bu aşama esnasında, aday tasarımlar, ihtiyaç özellikleri ile karşılaştırılmak üzere analiz edilmektedir. Simon, bu tasarım aşamasının önemini "*tasarım aşamasındaki en büyük çaba genellikle, önerilen alternatif çözümlerin test ve analiz edilmesine ayrılmaktadır*" diyerek vurgulamaktadır.



Dixon [14] ise tasarım prosesinin içinde "Mühendislik Analizi" denilen ve problemi tanımlamayı, model oluşturmayı, numerik sonuçları hesaplamayı, kontrol etmeyi, değerlendirmeyi, genelleştirmeyi ve sonuçları optimize etmeyi hedefleyen bir aşamanın var olduğunu kabul etmektedir.

Woodson [9] mühendislik analizini, önerilen bir tasarımın özellikleriyle uğraşan ve "Tahmin etme ve önem sırası analizi" diye adlandırdığı bir aktivite olarak tanımlamaktadır.

Tasarım problemine sistematik bir perspektiften yaklaşan Pahl & Beitz [2] ise, kesin olarak tasarım prosesinin içerisinde, performans açısından tasarımın fonksiyonel uygunluğunun değerlendirilmesi ile ilgili bir analiz aşamasının olduğunu söylemektedir. Aynı zamanda, bir mühendislik analizi prosesinin de kullanılan teknik ve metodolojik yardımlarla karakterize edildiğini belirtmektedirler.

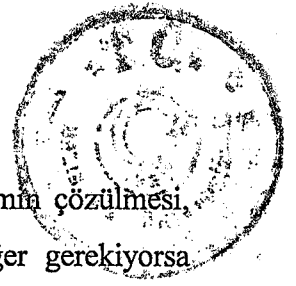
Beakley ve Chilton [15] da analiz prosesinin tasarım prosesi içinde özel bir aşama olduğunu kabul etmektedirler. Onlar analiz aşamasını bir tasarımın performansını, dayanıklılığını, ve davranışını hesaplamada kullanmaktadırlar.

### **2.3.2 Mühendislik Analiz Prosesi**

Mühendislik analizi çok adımlı bir prosestir ve Shepard ve arkadaşlarına [16] göre de bu adımlar aşağıda sıralandığı gibidir;

1. Fiziksel problem için uygun bir matematiksel modelin belirlenmesi,
2. Spesifik problemin analiz özelliklerinin ve alanın tanımlanması,
3. Spesifik problem için ek idealleştirmelerin gerçekleştirilmesi,
4. Nümerik analize uygun problemin açık ifadesinin oluşturulması ve temel fonksiyonların seçilmesi,





5. Problemin münferit elemanlara ayrılması, düzenlenmesi ve problemi çözülmesi, elemanlara ayırma hatalarının yeniden gözden geçirilmesi, ve eğer gerekiyorsa elemanlara ayırma işleminin iyileştirilerek yeniden yapılması ve yukarıda adı geçen adımların tekrarlanması,
6. Yapılan idealleştirmelerin (yani bir takım kabullerin) yeniden değerlendirilmesi, eğer gerekliyse 1,2 veya 3 nolu adıma hangisi uygunsa geri dönülmesi,
7. Sonuçların simule edilmesi ve sunulması.

Yine aynı çalışmada [16], analiz edilecek olan modelin münferit elemanlara ayrılmasının ve bir takım kabuller yapılmasının önemine işaret edilmekte ve idealleştirme yani bir takım kabuller yapma ve münferit elemanlara ayırmanın tanımları yapılmaktadır. Aşağıda bu tanımlar alıntı olarak verilmiştir;

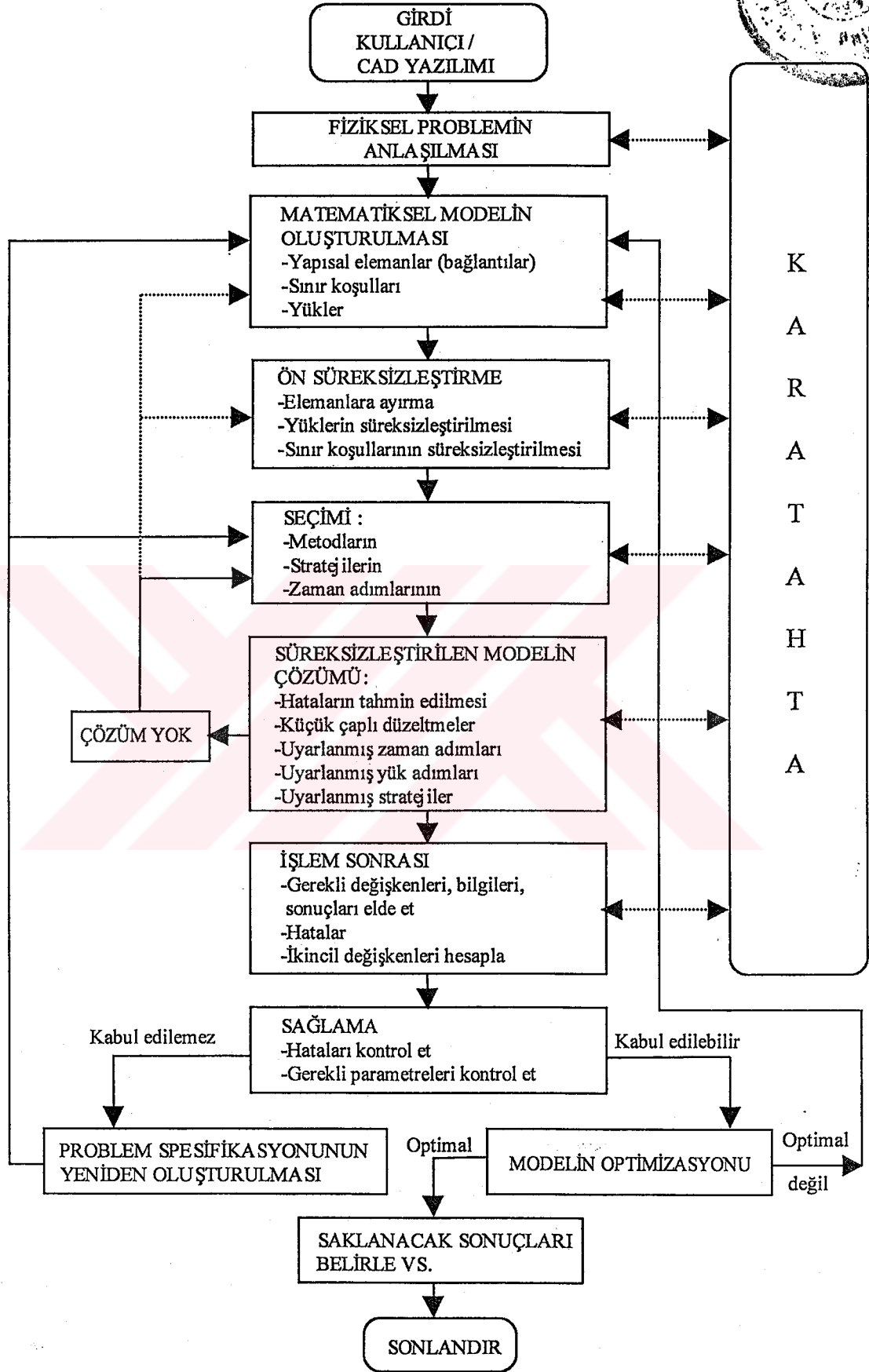
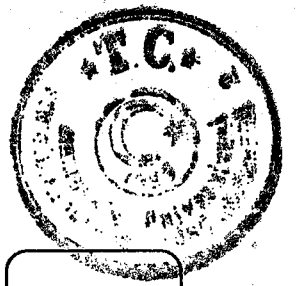
"idealleştirme, fiziksel bir olayın, analiz prosesi için yararlı olabilecek bir tarzda temsil edilmesidir. Münferit elemanlara ayırma ise, problemi en iyi ifade eden ve sadece sonlu büyüklükte ve sayılabilecek kadar sonsuz elemanlara ayırmaya izin veren sunumun seçilmesidir."

Tworzydło ve arkadaşlarının [5] sundukları bu çalışmada ise, sonlu elemanlar yöntemine dayalı, hesapsal mekanik alanında yapılan, analiz prosesini otomatik hale getirme konusuna yer verilmiştir. Bu makalada önerilen ve mühendislik analiz işlemini oldukça detaylı bir şekilde anlatan analiz prosesi Şekil 2.5 de verilmiş ve aşağıda da ayrıntılı olarak anlatılmıştır.

Proses "**Fiziksel problemin anlaşılması**" ile başlar. Bu aşamada, bir yapının bütün kendini diğerlerinden ayıran özellikleri tanımlanır ve tasarımın amaçları belirtilir. Bu aşamada verilen kararlar genellikle sezgisel veya tasarımcının önceki deneyimlerine dayalıdır. Bir sonraki adım ise o yapıya ait "**Matematiksel modelin oluşturulması**" dır. Bu adımda ;

- analizi yapılacak alanın tanımlanması, ve





Şekil 2.5 Tworzydło'ya göre mühendislik analiz prosesi [5]



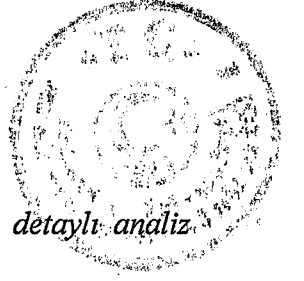
- yapının fiziksel davranışını en iyi şekilde temsil edebilecek matematiksel formülasyonun seçilmesi,

işlemleri yapılır.

Bir sonraki aşama, *“elemanlara ayırma”* yani sınır koşullarının, yüklerin ve analize tabi tutulacak alanın elemanlara ve düğüm noktalarına ayrılması suretiyle bölünmesidir. Sonraki adım ise, *“Hesapsal metodların ve stratejilerin seçilmesi”* dir. Verilen bir problem için kullanıcı özel bir metod seçmek zorundadır. Bu seçim işlemi, hatırı sayılır derecede bir uzmanlığa ve kullanıcının deneyimine gereksinim gösterir. Bunu izleyen adım ise, nümerik bir metod kullanılarak *“Elemanlara ayrılmış modelin çözümü”* dir. Bu aşama en çok hesaplanan yapıldığı bir evredir. Daha sonra ise *“bilgisayarda elde edilen bilgilerin değerlendirilmesi”* aşaması gelmektedir. Burada sonuçlar anlaşılabilir bir biçimde veya hesaplanmış yeni fonksiyonlar, örneğin gerilme, yüzey gerilmesi, yada kullanıcının istediği bir değer, şeklinde sunulurlar. Bu aşama aynı zamanda çözümün kabul edilebilir veya reddedilebilirliği ve girdi bilgilerinin değiştirilmesi hakkında gerekli bilgileri de temin eder. Çözümün ve model değişikliğinin *“doğruluğunu sağlama”* aşamasındaysa çözüm, tasarım kriterlerine ve amaçlarına göre bir kez daha kontrol edilecektir. *“Sonuçların saklanması”* adımı da eğer çözüm kabul edilebilir ve optimum ise gelecekteki kullanımlar için bu sonuçlar saklanacaktır.

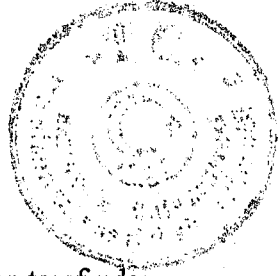
### 2.3.3 Mühendislik Analizinin Sınıflandırılması

Mühendislik analizinin karmaşıklığı tasarım prosesi boyunca farklılıklar göstermektedir. Tasarımın ilk safhalarında, tasarımın tasviri tam ve kesin değildir. Bu durum ise nümerik analiz araçlarının kullanılmasına izin vermez. Böyle bir durumda yapılan bir analiz ise yaklaşık analiz olarak adlandırılabilir. Tasarımın daha sonraki aşamalarında, tasarımın yapısı daha açık ve ayrıntılıdır ve genellikle Sonlu Elemanlar Analizi gibi nümerik analiz metodları sık sık kullanılır. Bu tür bir analize detaylı analiz de denebilir.



Finn [17] mühendislik analizini, *niteliksel analiz*, *yaklaşık analiz*, ve *detaylı analiz* olmak üzere üç kategoriye ayırmaktadır.

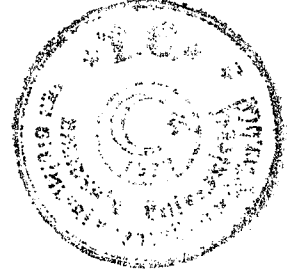
- **Niteliksel Analiz** zihinsel bir faaliyet olarak da tanımlanabilir. Bu tür analizde analizin düzeyi, mühendislik tecrübesinden ve sezgilerden tutun mühendislik kanunlarına ve ilkelerine kadar değişebilir. Niteliksel analiz genellikle bir fiziksel sistemin işlevinin veya davranışının anlaşılmasını sağlar.
- **Yaklaşık Analiz** ise ek nicel bilgi sağlar fakat, detaylı analizle ilgili olan, örneğin simülasyon gibi, bir takım karmaşıklıklardan da uzak kalır. Yaklaşık analiz bir yandan, önceden tahmin edilen ve beklenen davranış biçimi arasındaki uyumsuzluğu belirlemek için yeterli bilgiyi sağlarken diğer yandan da tasarım fonksiyonunun ve beklenen davranış biçiminin yeniden tanımlanması suretiyle tasarımın yeniden formüle edilmesine de olanak verir. Yaklaşık analiz, genellikle kağıt üzerinde gerçekleştirilir, bir takım mühendislik kabullerine ihtiyaç gösterir, ve niteliksel analizden daha çok zaman alan (örneğin ısı transfer probleminin Newton'un konveksiyon kanunu yardımıyla analizi gibi) mühendislik yaklaşım bilgisine gereksinim duyar.
- **Detaylı Analiz** nümerik bir simülasyon olarak da sınıflandırılabilir. Bu analiz, analiz edilecek bir alan hakkında yüksek hassaslık derecesinde kesin çözümler sağlamak üzere yapılır. Detaylı analiz, analiz yapan kişinin isteklerine bağlı olarak çeşitli hassasiyet düzeylerinde yapılabilir. Genellikle, detaylı analiz tasarımın son aşamalarıyla ilişkilidir, ama bazen basit nümerik modeller de göreceli olarak daha az bir çaba ile yaklaşık çözümler verebilirler. Detaylı analiz, aynı yaklaşık analizde olduğu gibi bir tasarımın yeniden formüle edilmesi için gerekli bilgiyi sağlamaktadır.



## 2.4 Bölümsel Sonuç ve Değerlendirmeler

Bu bölümde, mühendislik tasarımı konusunda çalışan çeşitli araştırmacılar tarafından yapılan mühendislik tasarımına ait tanımlar verildi ve tasarım prosesinin aşamaları değerlendirildi. Devam eden kısımlarda ise bu kez mühendislik analizinin tanımları verilir, analizde yer alan aşamalar ve bu aşamalarda yer alan aktiviteler değerlendirildi.





### 3. ÜRÜN MODELLEME

#### 3.1 Giriş

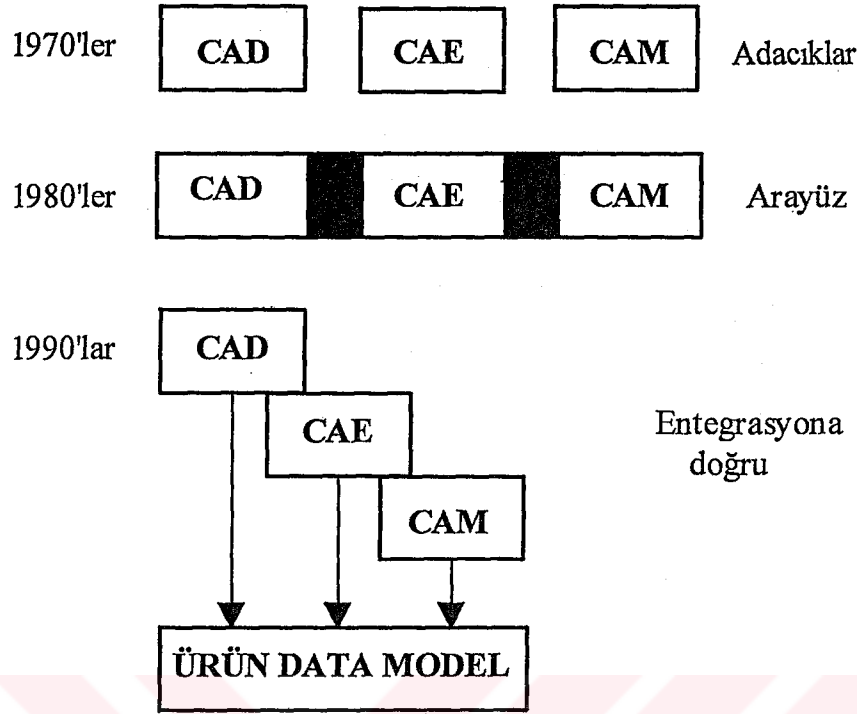
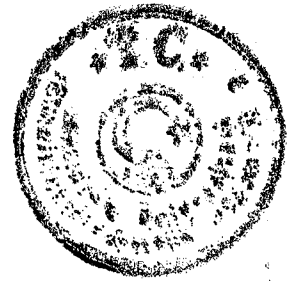
Bu bölümde bilgisayar destekli mühendislik faaliyetlerinin entegrasyonunun izlediği trend, geometrik modelleme, Leeds ve Loughborough üniversitelerinin işbirliği ile geliştirilen ve ürün veri modelleme diye adlandırılan kavram, yapısal veri tabanı editörü ve bu alanlarda yapılan çalışmalar gözden geçirilecektir.

#### 3.2 Bilgisayar Destekli Faaliyetlerin Entegrasyonu

Son yıllarda, verimliliği arttırmak ve bir ürünün piyasaya çıkarılma zamanını kısaltmaya yönelik CAD, CAM, CFD gibi çok sayıda bilgisayar destekli yazılım paketleri mevcuttur. Fakat bu bilgisayar destekli faaliyetlerin potansiyellerinden, bu aktiviteleri birbirleriyle entegre etmeden tam olarak yararlanmak mümkün değildir.

1970'li yıllarda, bilgisayar destekli aktiviteler, örneğin CAD, CAE ve CAM gibi, birbirlerinden bağımsız olarak geliştirildiler ve bu da "islands of information" yani bilginin adacıklar halinde öbeklenmesi denilen bir problemi doğurdu. 1980'li yıllarda ise eğilim bu aktiviteler arasında arayüzler oluşturma yönündeydi. Fakat 1990'larda, artık amaç bunları birbirlerine entegre etmektir. Bilgisayar destekli aktivitelerin entegrasyonu ile ilgili eğilimi gösteren bir yapı, Şekil 3.1 de gösterilmiştir.

Bilgisayar destekli aktiviteleri birbirlerine entegre etmenin yollarından birisi bu aktiviteler arasında veri alış-verişini temin etmektir. IGES, PDES ve son yıllarda STEP veri değişim standartları bu amaca hizmet etmektedir. Fakat, bir ürüne ait



Şekil 3.1 Bilgisayar destekli aktivitelerin integrasyon trendi [17]

verilerin birden çok kez üretilip kullanılıyor olması ve bilgisayar destekli aktiviteler süresince bu verilerin birbirlerine olan uygunluğunun kontrol edilmesi hala bir problem olarak durmaktadır. Bu probleme gerçek çözüm ise, bu çalışmanın ilerleyen bölümlerinde açıklanacak olan merkezi bir ürün veri modeli (Product Data Model) oluşturulmasının altında yatmaktadır[19].

Ürün veri modelinin (Product Data Model) açıklanmasına geçmeden önce, bu tür bir modellemenin gerekliliğine götüren ve ürün veri modellemesinin bir önceki aşaması olan Geometrik Modelleme (Geometric Modelling) kavramının kısaca bir gözden geçirilmesi önemlidir.

### 3.3 Geometrik Modelleme

Geometrik modelleme terimi, bir nesnenin geometrik özelliklerinin tam bir



matematiksel tanımlamasını yapabilecek metodların bir araya getirilmesi olarak tanımlanmaktadır[20].

Şu anda mevcut olan geometrik modelleme yöntemleri ise üç tanedir. Bunlar telkafes (wireframe) modelleme, yüzey (surface) modelleme ve katı (solid) modelleme yöntemleridir. Telkafes modelleme en eski 2 ve 3 boyutlu geometrik modelleme yöntemidir. Günümüzde kullanılanlar ise yüzey ve katı modelleme yöntemidir.

Bir nesnenin yüzey modelleme tekniği ile yapılmış modeli, telkafes yöntemi ile yapılmış olan modeline göre daha eksiksiz ve daha az belirsizlikler içerir. Yüzey modelleme tekniği genellikle havacılık sanayiinde, gemi tasarım ve inşaatında ve karmaşık dış hatlara haiz parçaların üretildiği otomotiv sanayiinde kullanılmaktadır.

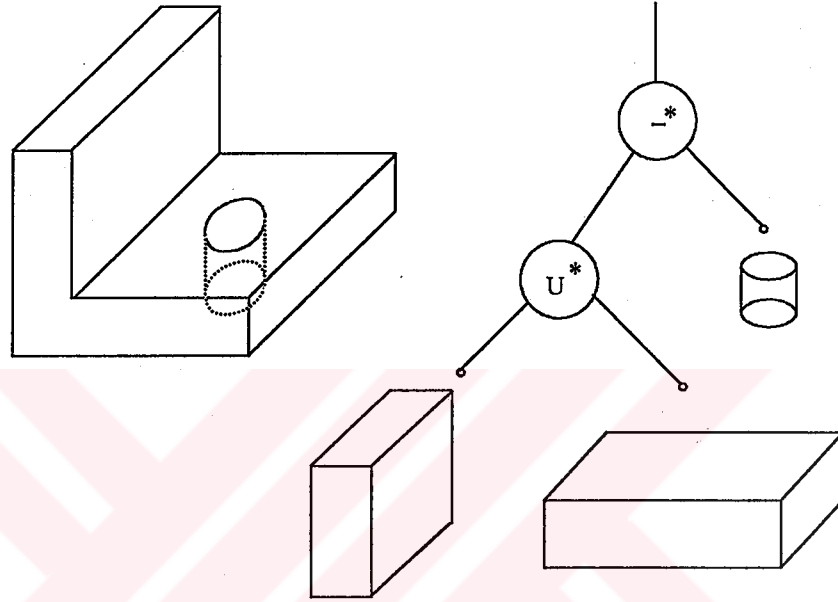
Geometrik modellemenin bir dalı olan katı modelleme ise katı bir nesnenin özelliklerini, örneğin nesnenin kütlesi gibi, tanımlamayı amaçlamaktadır [21]. Katı modelleme alanındaki araştırmalar, ilk olarak 1960'lı yıllarda görülmeye başladı ve 1970'li yılların ortalarına doğru da ilk kuşak deneysel amaçlı sistemler ortaya çıktı. Bu ilk sistemler büyük ilgi uyandırdı, çünkü bu sistemler geleneksel olarak elle yapılan bir takım işlemlerin örneğin detay çizimleri, sonlu elemanlar analizi için elemanlara ayırma, NC (Numeric Control) programlarının doğruluğunu sağlama gibi, tamamen otomatik hale getirilebileceği konusunda bazı vaatler sunuyordu [22, 23]. Sadece geometrik verileri içeren telkafes ve yüzey modellerden farklı olarak, katı modeller ilgili nesneye ait hem geometrik veri hem de topolojik bilgi içermektedir.

Şu anda piyasada bulunan ticari katı modelleme sistemleri CSG (Constructive Solid Geometry ) ve B-Rep (Boundary Representation)'dir. CSG yönteminde, bir nesnenin tanımlanması için, dikdörtgenler prizması, silindir, koni, küre, halka ve kama biçimindeki basit üç boyutlu nesnelere dayanılarak yararlanılmakta ve regüle edilmiş Bool işlemleri kullanılmaktadır [24, 25, 26].

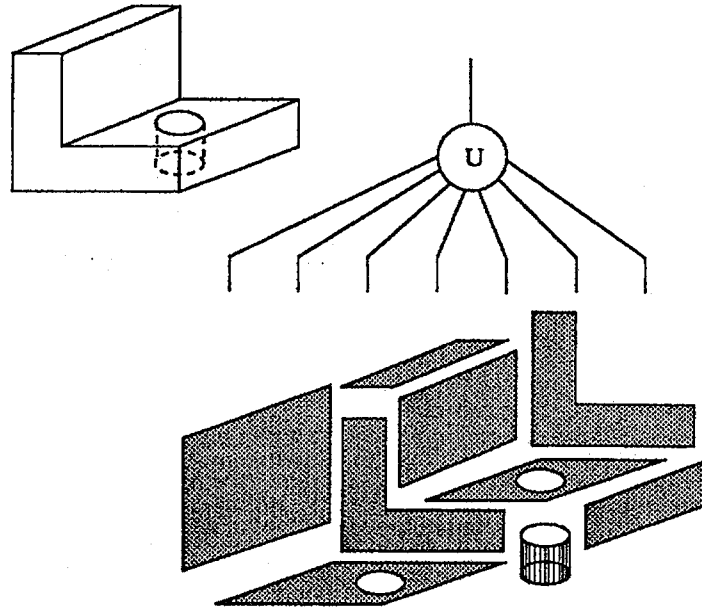
B-Rep yönteminde ise bir nesne onu çevreleyen sınırlarıyla tanımlanır. Katı bir



nesnenin sınırı, tepe noktaları, ayrıtlar ve yüzeylerle, bunların üç boyutlu Euclid uzayında  $[E^3]$  birbirleriyle olan ilişkileriyle tanımlanır. Örnek bir nesnenin hem CSG hem de B-Rep yöntemiyle tanımlanmış biçimi sırasıyla Şekil 3.2 ve Şekil 3.3 de sunulmuştur. Geometrik modelleme ile ilgili daha detaylı bilgi ise [27, 24, 21, 25, 22 ] nolu referanslarda bulunabilir.

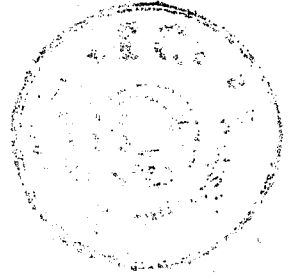


Şekil 3.2 Bir nesnenin CSG modelleme yöntemi ile ifadesi



Şekil 3.3 Bir nesnenin B-Rep modelleme yöntemi ile ifadesi





İngiltere'deki Leeds Üniversitesinde, geometrik modelleme ile ilgili proje 80'li yıllarda başlamış ve sonuçta NONAME diye adlandırılan bir katı modelleme sistemi geliştirilmiştir. NONAME baz olarak CSG yöntemini almış ve üç boyutlu basit nesne olarak ise dikdörtgenler prizması, silindir, küre, koni ve halka kullanılmıştır [28]. Bu çalışmanın devamında ise, imalatla ilgili uygulamaları otomatik hale getirmek için sadece geometrik bilgilerin yeterli olmadığı ve daha geniş kapsamlı diğer tipte bilgilerin de gerekli olduğunun farkına varılmış [19, 29] ve sonuç olarak Loughborough teknoloji üniversitesi ile işbirliğine gidilerek, Bilgi Destekleme Sistemleri (ISS) araştırma projesi çerçevesinde Structure Editor (SE) denilen ve veri tabanı editörü diyebileceğimiz, çalışmanın da ilerleyen kısımlarında açıklanacak olan, bir bilgisayar yazılımı geliştirilmiştir.

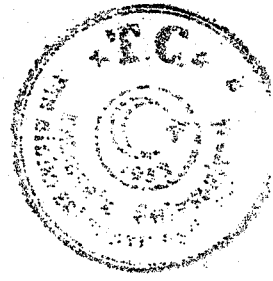
### **3.4. Leeds Üniversitesinde Ürün Veri Modellemesi**

Bu bölümde, Leeds ve Loughborough Teknoloji üniversiteleri tarafından imalatla ilgili uygulamaları entegre etmek amacıyla başlatılan ve ISS ( Information Support Systems for Design and Manufacture) diye adlandırılan araştırma projesi esnasında geliştirilen Ürün Veri Modelleme ortamı ana hatları ile anlatılacaktır.

Aynı zamanda, Ürün Modelleme (Product Model) kavramı, bu çalışmanın altıncı bölümünde verilecek olan CFD veri modelinin geliştirilmesine de temel teşkil edecektir.

#### **3.4.1 Ürün Veri Model Çerçeve Çalışması**

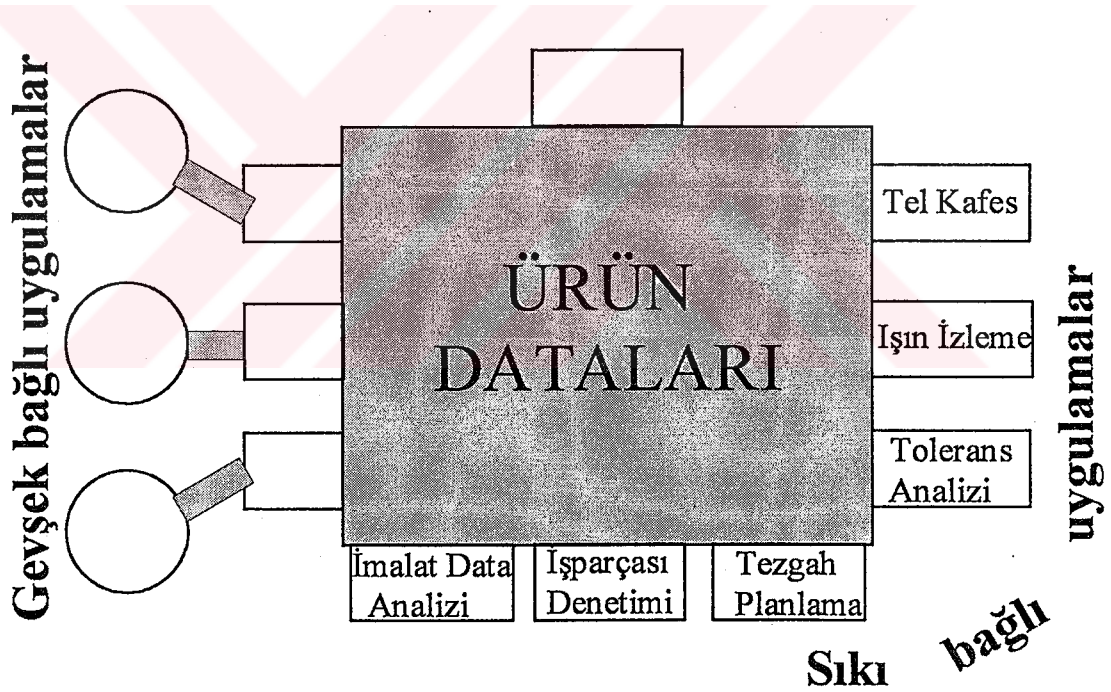
Ürün Modeli (Product Data) terimi, bir ürünün spesifikasyonlarından, tasarımına, ve üretimine kadar geçen ve ürün yaşam süreci boyunca elde edilen çeşitli mühendislik verilerinin bilgisayar ortamında temsil edilmesi olarak tanımlanabilir [30].



Ürün Veri Modeli üç koşulu sağlamaktadır;

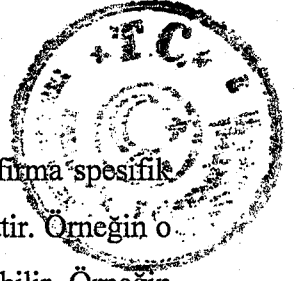
- birbirinden ayrı bilgisayar yazılım uygulamalarının entegrasyonunu destekler,
- yeni veri modelinin ilavesi ile oluşabilecek sonuçların önceden tahmin edilebilmesi için bir altyapısal düzenleme sağlar,
- her işletmenin ihtiyacına uygun olarak yeniden düzenlenebilir.

Şekil 3.4'de Ürün Veri Modelleme ortamının yapısı gösterilmiştir. Şekilden de görüleceği üzere, burada ürün verisi bu yapının özünü teşkil etmektedir. Çeşitli mühendislik uygulamaları ise bu ürün verisiyle entegre edilmiş durumdadır [19].



Şekil 3.4 Ürün veri modeli mimarisi

Leeds üniversitesinde geliştirilmiş olan ürün veri modeli iki farklı veri türünü desteklemektedir. Birincisi, bilgisayar yazılımı tarafından kullanılacak olan veri, ki bu tür verinin yapısı mutlaka önceden belirlenmiş olmalıdır, ve ikincisi ise kullanıcı tarafından ihtiyaç duyulabilecek olan veridir ki bu tip bir veri yapısı ise kişilerin veya



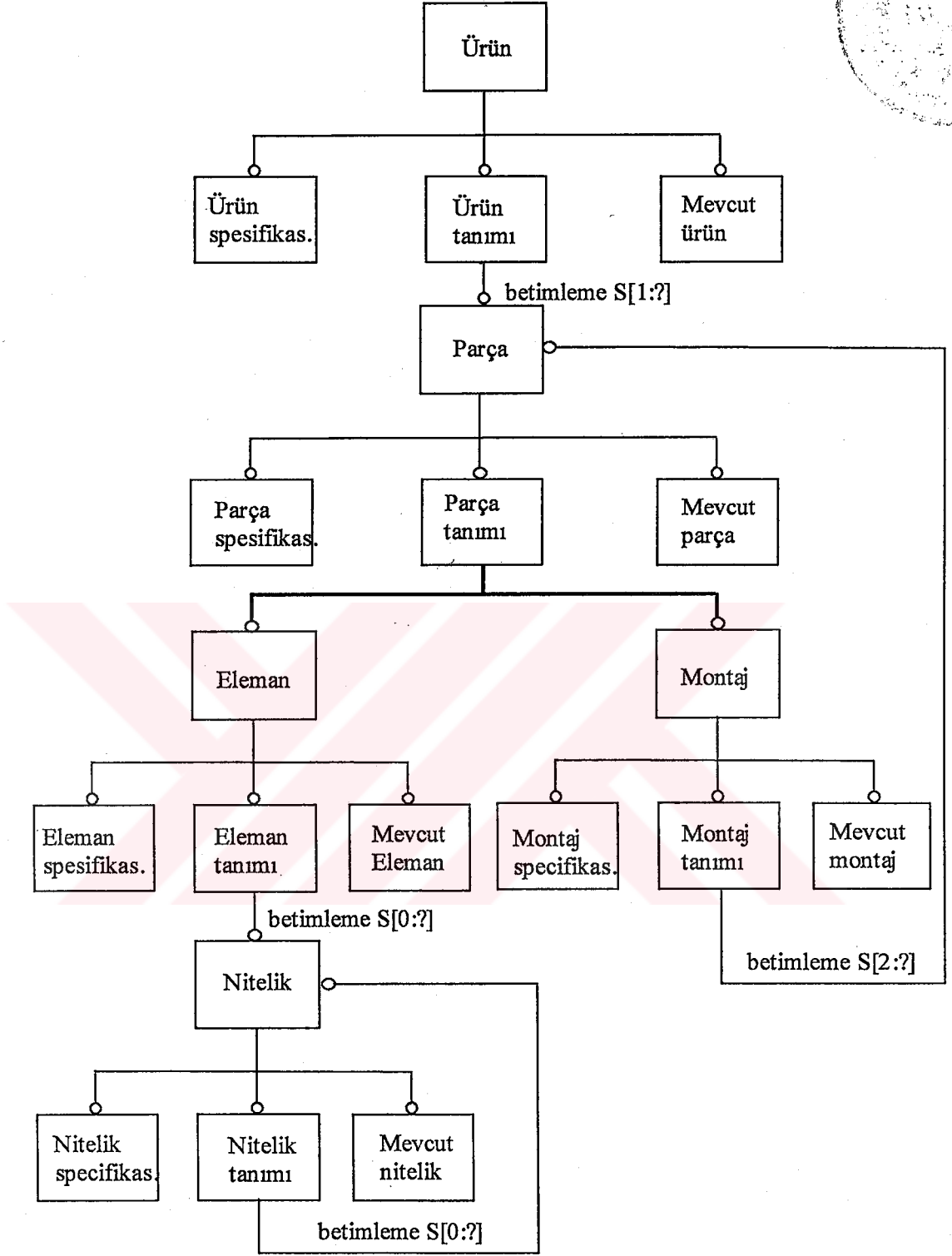
işletmelerin ihtiyaçlarına göre değişebilir. Bu veriler ise ürün spesifik, firma spesifik veya referans veri türünde olabilir. Ürün spesifik veri sadece bir ürüne aittir. Örneğin o ürünün geometrisi gibi. Firma spesifik veri ise birden fazla ürüne ait olabilir. Örneğin bir şirketteki farklı ürünleri üretmek için kullanılacak olan üretim imkanlarına ait veriler bu tür veriye bir örnektir. Referans veri ise çok daha geneldir. Örneğin standartlarda yer alan ve bir malzemeye ait bir takım özellikler ve veriler gibi [31].

Şekil 3.5’de elektro mekaniksel ürünlerin tanımlanmasında kullanılan bir ürün veri modelinin çerçeve çalışması görülmektedir[32]. Çerçeve çalışma EXPRESS-G modelleme dili kullanılarak yapılmıştır[33]. Bu dille ilgili açıklama ise Ek-B’de bulunabilir.

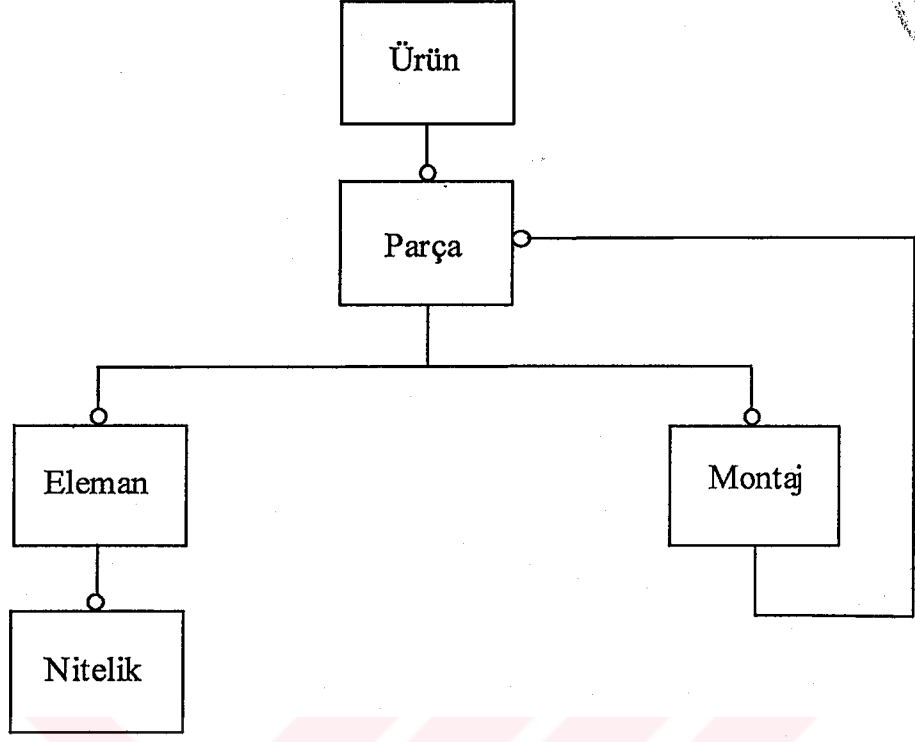
Bu çerçeve çalışma, ürün verisini iki farklı kategoriye ayırmaktadır. Bu kategorilerin herbiri, ürün yaşam sürecinin farklı yanlarını ifade etmektedir. Bu kategoriler ise “bir ürün içindeki düzeyler” (Levels within a product) ve “tekrar eden kalıp” (Repeating pattern) olarak adlandırılmaktadır. Bu farklı hiyerarşik düzeylerin ve tekrar eden kalıpların kombinasyonu ise spesifik uygulamaların desteklenmesini sağlamaktadır.

- Düzeyler :

Bir ürünü oluşturan farklı hiyerarşik düzeyleri şu şekilde sıralamak mümkündür; ürün, parça, eleman, eleman ve/veya parçalardan oluşan grup(assembly), ve niteliklerdir. Şekil 3.6’da bir ürünün içindeki farklı hiyerarşik düzeyler görülmektedir. Bir ürün bir yada birden fazla parçadan oluşmaktadır. Bir parça ise, sadece bir elemandan müteşekkil olabileceği gibi birden fazla elemanın bir araya geldiği bir elemanlar grubundan da oluşabilir. Bir eleman ve/veya parçalardan oluşan grup (assembly) ise en azından iki veya daha fazla elemana sahip olmalıdır. Her bir eleman ise bir veya birden fazla niteliğe sahiptir.



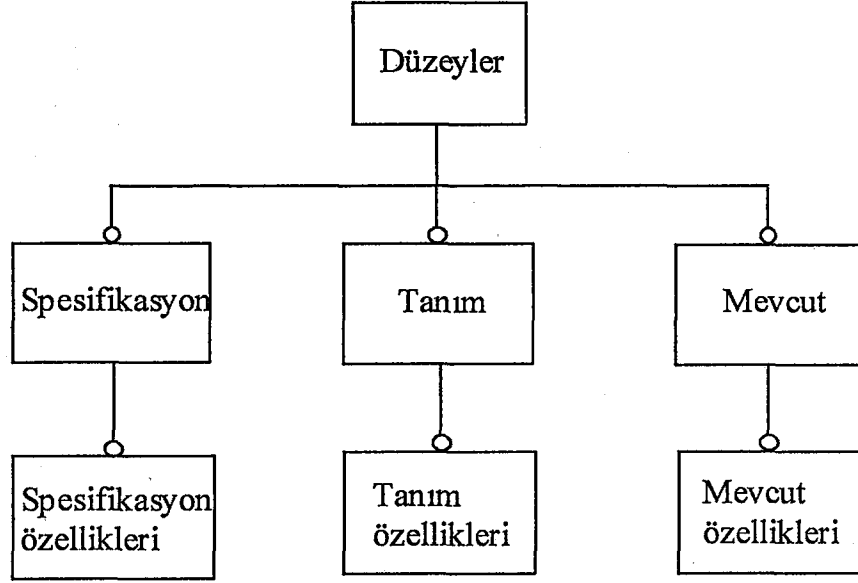
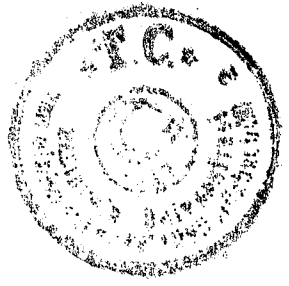
Şekil 3.5 Ürün veri modeli çerçeve çalışması



Şekil 3.6 Ürün veri modelindeki düzeyler

- Tekrar Eden Kalıp :

Şekil 3.6'da da gördüğümüz gibi, bir ürünün içerisinde beş farklı hiyerarşik düzey bulunmaktadır. Ve her düzey de bir ürünün yaşam sürecinin farklı yönleriyle ilişkili üç değişik tekrar eden kalıba sahiptir. Şekil 3.7 ye bakıldığında tekrar eden kalıplar daha iyi anlaşılabilir. Her tekrar eden kalıp bünyesinde bir spesifikasyon, bir tanımlama ve bir de güncel denilen üç unsur bulundurur. **Spesifikasyon** mutlaka sağlanması gereken koşulları tanımlamaktadır. Ve her ürün mutlaka bir spesifikasyona sahip olmalıdır. Tanımlama ise bir spesifikasyonun hangi yolla sağlanabileceğini belirler. Güncel ise hali hazırda fiziksel olarak mevcut olan bir nesneye ait verileri içerir[34].



Şekil 3.7 Ürün veri modelindeki tekrar eden kalıplar

### 3.4.2 Ürün Veri Modelini Hazırlamak İçin Bir Bilgisayar Yazılımı

Ürün Veri Modelleme ortamını etkileşimli olarak düzenlemek ve iyileştirmek için bir ürün veri editörünün gerekliliği, [19, 35] nolu referanslarda da belirtildiği gibi aşıkardır. Bu yüzden de Leeds üniversitesinde yukarıdaki ihtiyacı karşılamak üzere SE (Structure Editor)[36, 37] denilen bir bilgisayar programı geliştirilmiştir. Aşağıda da bu yazılım hakkında detaylı bilgi verilmiştir.

### 3.4.3 Structure Editor

SE, veri yapılarının oluşturulması ve oluşturulan bu veri yapıları ile uygulamalar arasında arayüzler oluşturulmasında kullanılan bir bilgisayar programıdır [38, 39, 40].

SE veri editörü ile veri yapısı üzerinde aşağı, yukarı, sağa, sola hareket etmek, yani bu yapı içindeki her noktaya ulaşmak mümkün olmaktadır. Bu yazılım sayesinde hiyerarşik, çevrimsel veya paylaşılan verileri edit etmek mümkün olmaktadır. SE çalıştırıldığında ise iki yapıya ihtiyaç duymaktadır. Bunlar, kontrol sağlayan bir şablon



yapı (meta-structure) ve bu yapının bir ürün için örneklendiği ve verilerinin girildiği bir örnek (the instance) yapısıdır.

Kontrolü sağlayan şablon yapı, kullanıcıya aşağıda içerikleri verilen 7 tip eleman sayesinde ürünün örneklenmesine yardımcı olur.

1. **Kolleksiyonlar** (Collections) belirli sayıdaki farklı tipteki elemanların bir kolleksiyonudur.
2. **Seçimler** (Selections) kullanıcının seçimdeki elemanlardan birini seçmesine imkan tanır.
3. **Listeler** (Lists) bir veya birden fazla aynı türdeki elemanlardan oluşur.
4. **Atomlar** (Atoms) yapıdaki dalların en ucunda bulunurlar ve daha alt elemanlara ayrılamazlar. Bunlar rakam atom, isim atom, dizin atom veya boş atom şeklinde olabilirler.
  - Rakam atomlar gerçel veya tam sayılardır.
  - İsim atomlar 16 karaktere kadar olan dizinlerdir.
  - Dizin atomlar 2000 karaktere kadar olan karakter dizinleridir.
  - Boş atomlar da karakter dizinleridir. Eğer bunlar şablon yapıda tanımlanmışlarsa örnek yapıda da görülmüştür.
5. **İsmlendirilmiş nokta** bir isim ve bir noktadan oluşur.
6. **Büyük Kolleksiyonlar** kolleksiyonların özel bir formudur, ama büyük kolleksiyonların düzenlenmesinin daha kolay yapılmasını sağlarlar.
7. **Soyutlanmış nokta** denilen kolaylık ise modüler kod üretimi işlemi sırasında kullanılmaktadır.

SE aynı zamanda otomatik kod üretimi diye adlandırılan ve ürün veri modelinden, veri yapısı ile uygulama programı arasında bir arayüz oluşturmak üzere, gerekli verileri çekip onları ADA programlama diliyle uygun biçime dönüştürerek bir arayüz oluşturma özelliğine de sahiptir [41]. ISS araştırma projesi çerçevesinde bu tür arayüzler de gerçekleştirilmiştir. Örneğin bir geometrinin boyutları ve toleransları ile ilişkili “boyutlar ve toleranslar” arayüzü[42], ve bir ürün için proses planı ve NC kodu üreten “üretim planlama” arayüzü gibi[43].

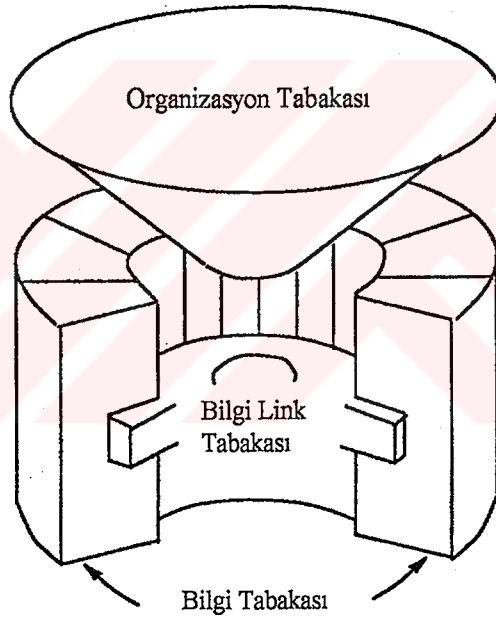




### 3.5 Ürün Modelleme ile İlgili Çalışmaların Gözden Geçirilmesi

Bu bölümde ürün modelleme konusunda yapılmış olan çalışmalar [44, 45, 46, 47, 48, 49] ele alınacak ve özetlenecektir.

Spur ve arkadaşları [47] Şekil 3.8 de verilen, ve bir şirketteki bütün bilgisayar yazılımlarını entegre etmeyi amaçlayan bir ürün modeli önermektedirler. Aynı zamanda onlar bir şirket içerisindeki farklı bölümler arasındaki bilgi akışının entegrasyon için elzem olduğunu vurgulamaktadırlar.



Şekil 3.8 Spur'ün ürün modeli [47]

Önerdikleri ürün modeli üç katmandan oluşmaktadır: Organizasyon katmanı, Bilgi katmanı ve Bilgi bağlantı katmanı. Bilgi katmanları, geometri gibi, proses planlaması ve benzeri gibi ürün verisini temsil etmektedir. Bilgi bağlantı katmanı da aynı ya da farklı bilgi katmanlarındaki bilgileri birbirine bağlar. Organizasyon katmanı ise nesnenin mantıksal görünüşünü temsil eder. Organizasyon katmanı bilgi katmanlarına

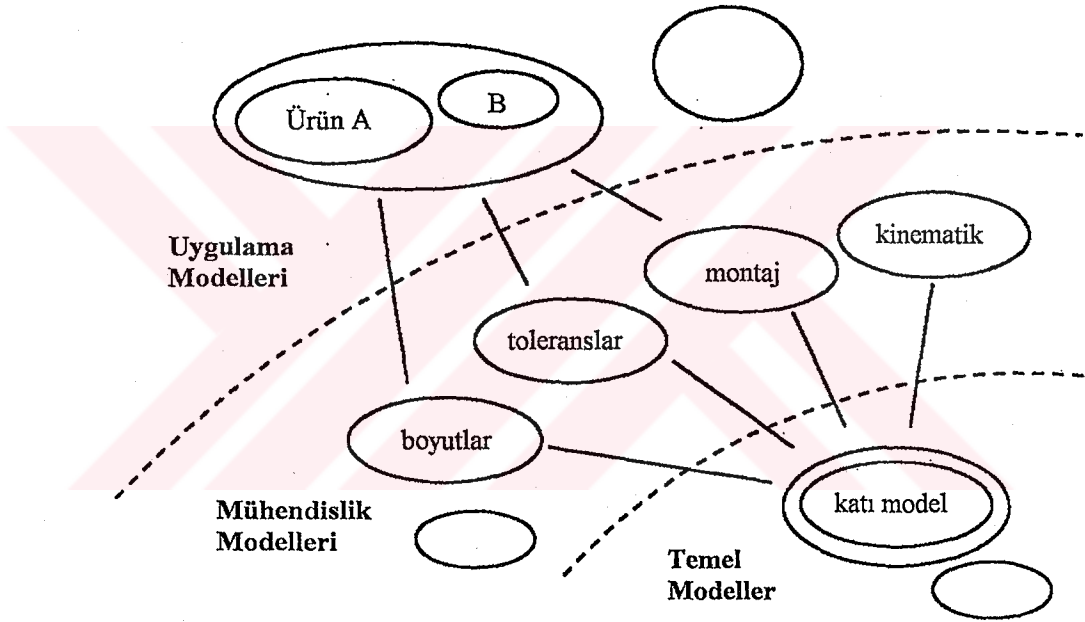


ve aynı zamanda da diğer ürün modellerinin organizasyon katmanlarına ait referansları da içermektedir.



Bilgi depolama ihtiyacını azaltmak ve gereksiz bilgi fazlalığından sakınmak amacıyla kendi ürün modellerinde veri paylaşımı kavramını dikkate almışlardır.

Sata ve Kimura ise [44], ürün modeli için nesne yönelimli bir yaklaşım ileri sürmektedirler. Makina imalatı için olan ürün modeli, boyutlar, toleranslar ve montajla ilgilidir. Ürün modellerindeki verileri üç değişik kategoriye ayırmaktadırlar; temel modeller, mühendislik modelleri, ve uygulama modelleri (Şekil 3.9'a bakınız).

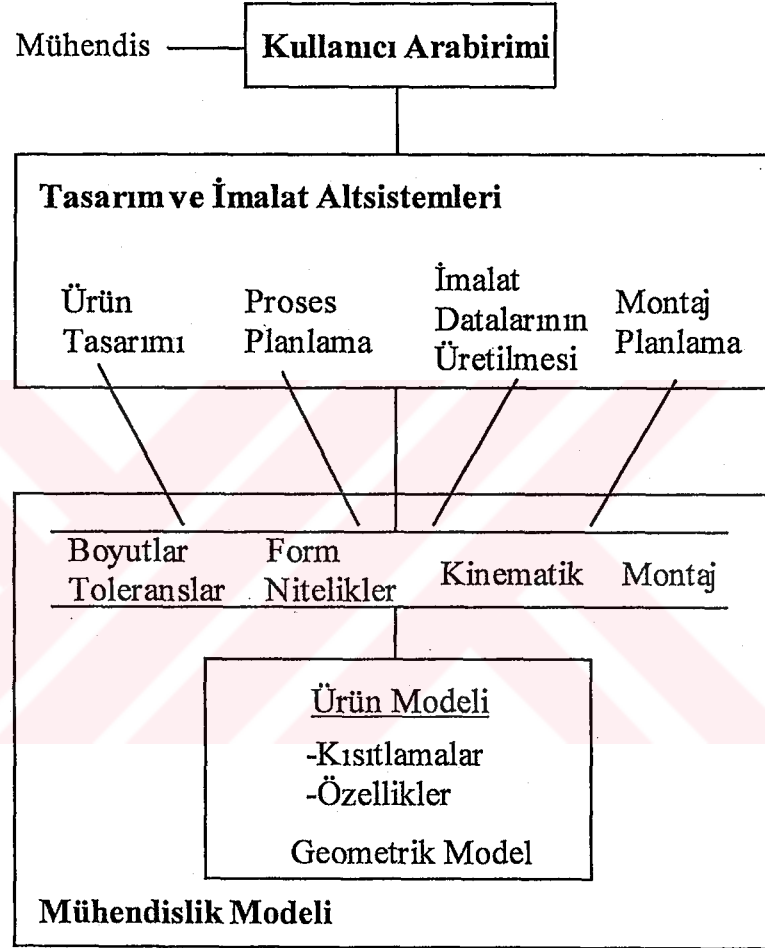


Şekil 3.9 Makina imalatı için ürün modeli [44]

Temel modeller, şekil temsili, sembolik matematiksel formüller, sonlu eleman ağ modelleri gibi bir nesne hakkındaki kısıtlamaları ve temel özellikleri ifade etmektedir. Mühendislik modelleri ise boyutlandırma ve tolerans, montaj ve kinematik ilişkiler, malzeme ve imalat metodları gibi temel mühendislik bilgilerini ifade etmektedir. Uygulama modelleri de, benzer ürünlerin modellenmesine yardımcı olmada kullanılacak hali hazırda tanımlanmış olan ürünlerin modelleridir. Kimura ve arkadaşları daha sonraki çalışmalarında ise [45, 50, 51], bir ürünün tasarımında daha

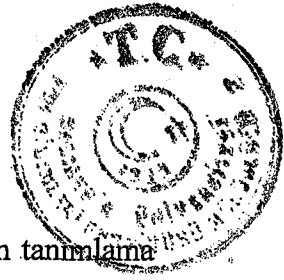


önceden tasarlanmış benzer bir ürün tasarımının baz alındığı fakat istenen yeni şartlara göre değiştirildiği, varyasyonel ürün tasarım kavramı üzerine yoğunlaşmışlardır. Ürün tasarımını, proses planlamasını, üretim planlama verilerinin üretilmesini ve montaj planlama uygulamalarını desteklemek amacıyla düşündükleri ürün modellerinin mimarisi Şekil 3.10 gösterilmiştir.

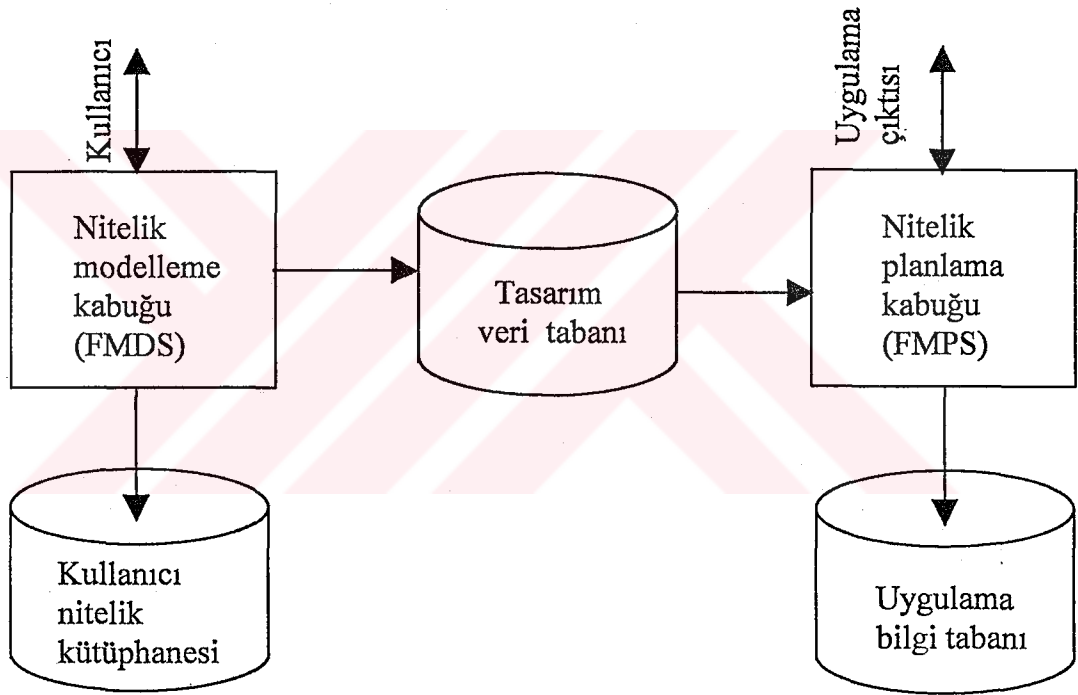


Şekil 3.10 Kimura'nın ürün modelinin sistem mimarisi [51]

Ürün modelleri üç ana elemanı içermektedir : Mühendislik modeli, Tasarım ve imalat alt sistemleri, ve kullanıcı ara birimi. Mühendislik modelleri, mekaniksel ürünler için farklı türdeki mühendislik bilgilerini temsil etmektedir. Tasarım ve imalat alt sistemleri uygulama görevini gerçekleştirirler. Örneğin ürün tasarımı, proses planlaması gibi. Kullanıcı ara birimi ise, çizimler ve diğer klasik sunumlar aracılığıyla, mühendislere bir iletişim ortamı sağlar.



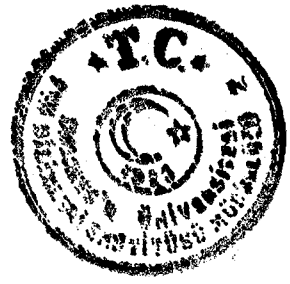
Shah ve arkadaşları [46] da, A.S.U. özellikler test yatağı denilen bir ürün tanımlama sistemi geliştirmişlerdir. Sistem, mekaniksel parçaların değerlendirilmesi ve dokümantasyonu, ve tasarım için modüllerin bir koleksiyonundan oluşmaktadır. Test yatağı iki kabuktan meydana gelmektedir (bakınız Şekil 3.11). Bunlardan biri tasarım kabuğu diğeri de planlama ve uygulamalar kabuğudur. Tasarım kabuğu özelliklerin, boyut ve toleransların, geometrinin, topoloji ve tasarım kurallarının ürün tanımlama sistemine entegre edilmesini sağlamaktadır. Özellik planlama kabuğu ise tasarım veri tabanını oluşturmak için ürün verilerini elde eder ve onları uygulama tarafından istendiği biçimde yeniden formüle eder.



Şekil 3.11 A.S.U.test yatağının sistem mimarisi [46]

Tasarım kabuğu yedi adet değişik modelleyiciyi kapsamaktadır. Bunlar sırasıyla;

- form özellik modelleyici,
- tolerans modelleyici,

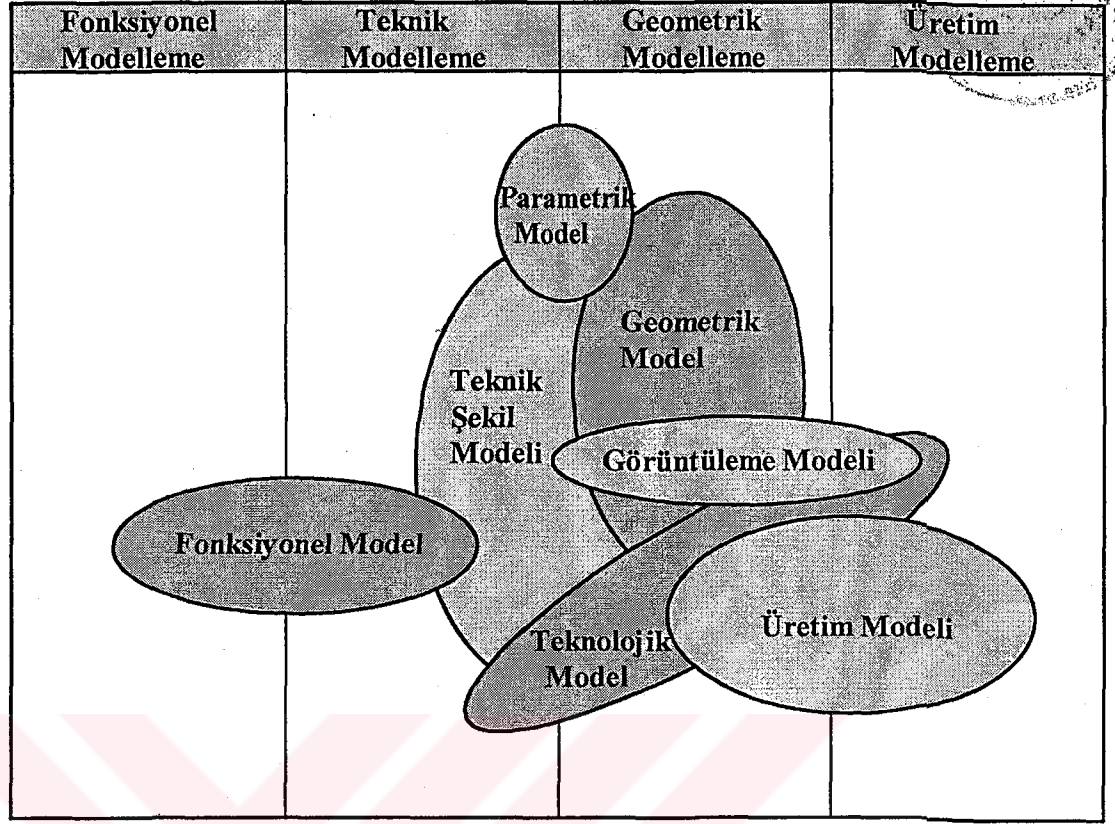


- malzeme modelleyici,
- etkileşimli/otomatik özellik tanımlama modülü,
- PDES ön ve son işlem birimleri,
- geometrik modelleyici ara birimi,
- pseudo-tabaka modelleyici.

Nitelik tasarlama kabuğu, imal edilebilirlik, gerilme analizi gibi bir takım uygulamaların yerine getirilmesine de yardımcı olur.

Entegre edilmiş Ürün Model kavramı, 1984 yılında, makina mühendisliği alanındaki bilgisayar destekli tasarım (CAD) ortamındaki bazı eksik ara birim özelliklerini araştırıp bulmak üzere başlatılan CAD Arabirimleri (CAD\*I) projesi sayesinde ortaya çıkmıştır[48]. Araştırma neticesinde de tasarım aşamalarını ifade etmenin iki yolu olduğu vurgulanmaktadır. Bunlardan birincisi, birkaç farklı modülün ardışık tarzda tanımlanması; diğeri de entegre edilmiş ürün modelidir. Entegre edilmiş ürün modeli Şekil 3.12 de görüldüğü gibi Venn diagramı ile ifade edilmektedir. Entegre edilmiş ürün modeli birkaç kısmi modeli içermektedir. Bunlar ;

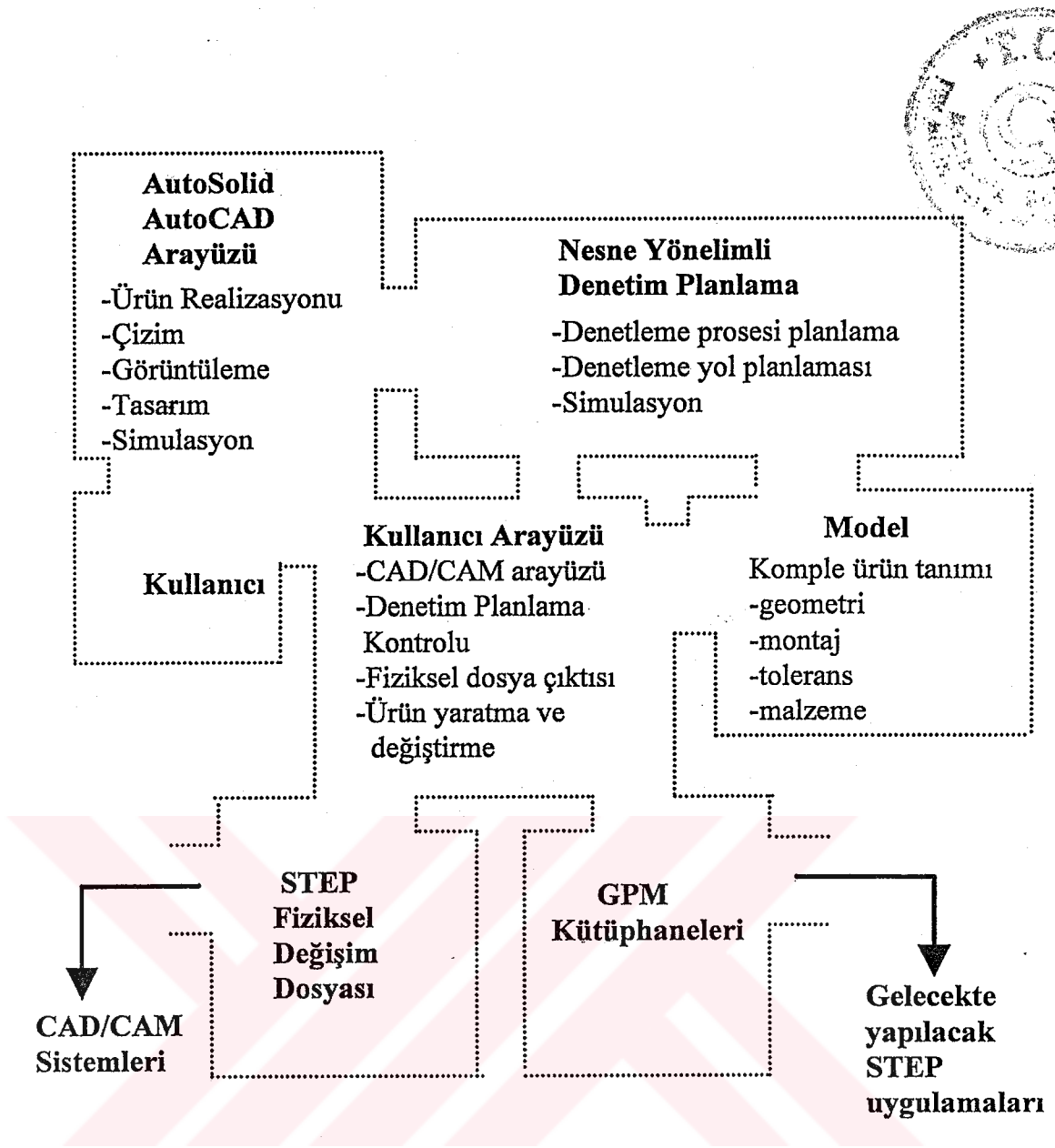
1. **Fonksiyonel Model** bir ürünün tüm fonksiyonlarını tanımlar ve onu alt fonksiyonlara ayırır,
2. **Teknik Şekil Modeli** bir ürünün kinematikini, yapı ve şekil bilgilerini ifade eder.
3. **Geometrik Model** bir ürünün geometrisini B-Rep modelleme tekniğini kullanarak tanımlar.
4. **Teknolojik Model** bir ürünün malzeme, yüzey yapısı, tolerans gibi teknolojik verilerini içerir.



Şekil 3.12 CAD\*I ürün modelinin mimarisi [48]

5. **Parametrik Model** ürün değişkenleri arasındaki matematiksel ilişkileri sağlar.
6. **Görüntüleme Modeli** grafiksel görünüşler ve mühendislik çizimlerini yaratmak için gerekli verileri içermektedir.
7. **Üretim Modeli** üretim ve takım verisi gibi imalat prosesiyle ilgili bilgileri içerir.

Gu ve arkadaşları [49] ise, standart veri formatı ve değişik imalat aktivitelerini entegre edecek tam bir ürün bilgi yapısını destekleyen, STEP tabanlı genel bir ürün modelleme (GPM-Generic Product Modelling) ortamı yarattılar. Şekil 3.13'de gösterilen ürün modelleme ortamları, bir kullanıcı ara birimi, bir denetim planlayıcı (inspection planner), bir AutoSolid ve AutoCAD ara birimi, ve ürün verisine kolayca ulaşılmasını sağlayan kütüphanelerden oluşmaktadır. Bu GPM sistemindeki kütüphaneler



Şekil 3.13 Gu'nun genel ürün modelinin mimarisi [49]

ise ÜrünveVersiyon (ProductandVersion), Ürünilişkileri (ProductDefnRelation), Malzeme (Materials), ŞekilTemsili (ShapeDefnRep), Geometrik Parça (Geometric Item), Tolerans (Tolerance) kütüphaneleridir. AutoSolid ve AutoCAD ara birimleri ise bilgisayar ekranında canlandırma, simulasyon, ve ürün tasarımı için uygulamaları desteklemektedir. ÜrünveVersiyon (ProductandVersion) kütüphanesi ürünleri ve onların değişik versiyonlarını içermektedir. Ürünilişkileri (ProductDefnRelation) kütüphanesi farklı ürünler arasındaki ilişkisel bilgileri bünyesinde barındırır. Malzeme (Materials) üretim prosesi için uygun malzeme listesini içerir. ŞekilTemsili (ShapeDefnRep) kütüphanesi ürünlerin şekilsel betimlemesini saklamaktadır. Geometrik Parça (Geometric Item) kütüphanesi de ürünleri tanımlayacak gerekli





geometrik parçaları içermektedir. Tolerans (Tolerance) kütüphanesi ise ürün tasarımı için gerekli tolerans bilgilerini saklamaktadır.

### 3.6 STEP Ürün Modeli Verilerinin Değişimi için Standart

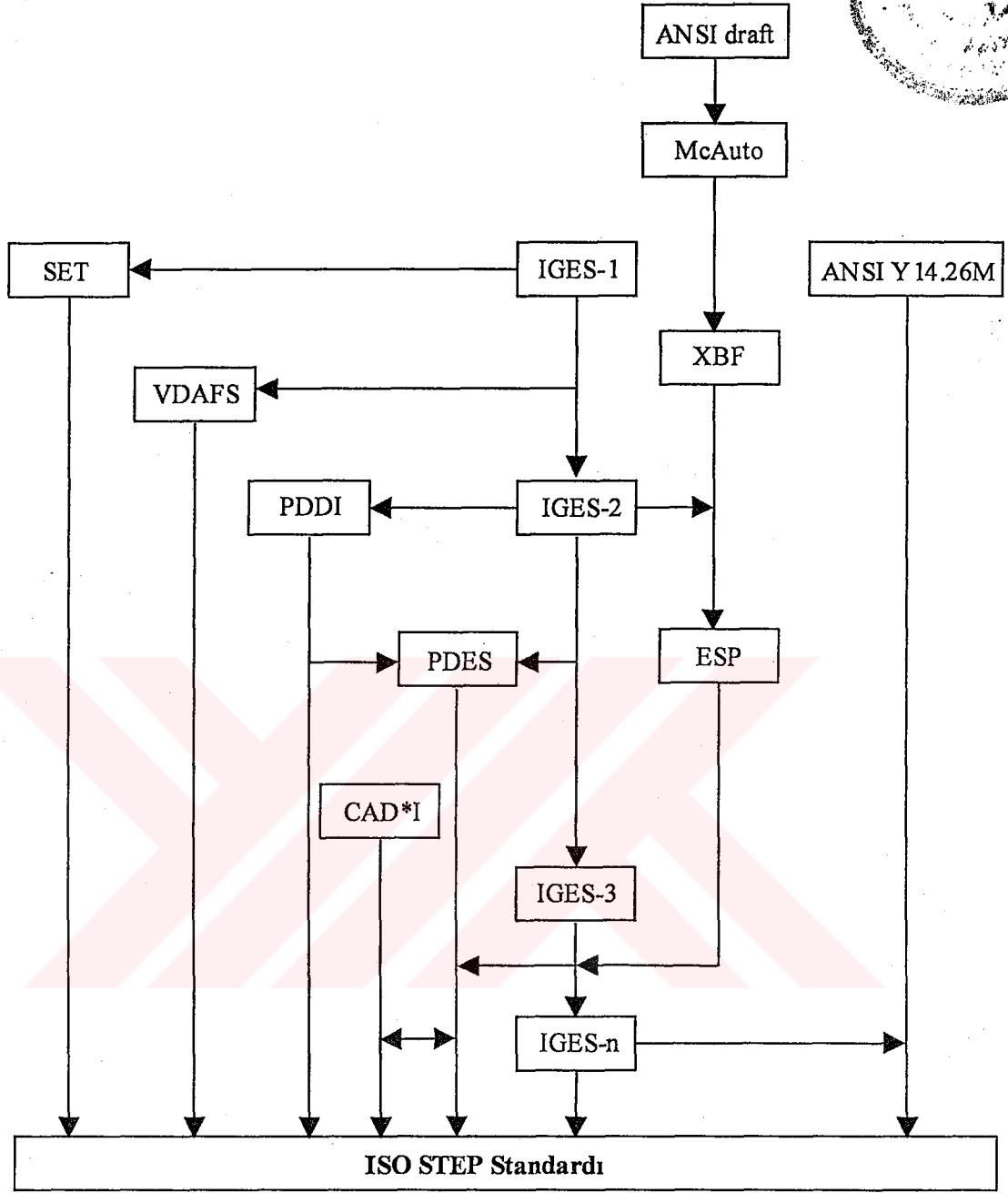
STEP (Standard for the Exchange of Product Model Data), yeni bir mühendislik ürün veri değişimi standardı geliştirmek üzere, uluslararası Standart Organizasyonu ISO tarafından yürütülmekte olan bir araştırmanın adıdır. Standardın amacı ise, çok sayıda farklı ulusal standartlara sahip olmak yerine, tüm endüstrilerde ürün yaşam sürecinin tüm yönlerini kapsayacak tek ve daha iyi bir standart sağlamaktır [52].

STEP'i ;

1. bir ürünü, o ürünün tasarlandığı aşamadan iskartaya çıktığı aşamaya kadar tanımlamak için gerekli olan bilgileri ifade etmenin standart bir yöntemi, ve
2. bu bilgileri ifade eden verilerin elektronik yolla değişimi için standart metodlar diye de tanımlamak mümkündür[53].

STEP projesi şu anda bulunduğu aşamaya gelinceye kadar daha önceden var olan bir çok farklı ulusal standarttan etkilenmiş olup bu standartların önemli olanları aşağıda kısa bir şekilde özetlenmiş ve Şekil 3.14'de de bu standardın etkilendiği standartlar ve projenin gelişimi ayrıntılı bir şekilde gösterilmiştir [90, 91]. Ayrıca STEP ile ilgili daha ayrıntılı bilgi ise, STEP'in etkilendiği diğer standartların bitiminden hemen sonra verilecektir.

1970'li yılların başlarında, verileri elektronik olarak çizim sistemleri arasında transfer etmenin gerekliliği ANSI Y14 diye anılan komite tarafından fark edildi. Ve bu konudaki ilk girişim transfere baz olarak alınabilecek bir dosya formatı standardının geliştirilmesi oldu. Yetmişli yılların sonlarına doğru ise, veri transferinin sadece çizim sistemleri arasında değil katı modelleme sistemleri arasında da olması gerektiği CAM-I



Şekil 3.14 Veri değişim formatının evrimi [53]

denilen geometrik modelleme projesi ile ortaya kondu. McDonell Douglas Automation (McAuto) projesi çerçevesinde de, ANSI çalışma dokümanlarındaki fikirler dikkate alınarak katı modelleme verileri için bir dosya şartnamesi hazırlandı. Kısa bir süre sonra da Shape Data Ltd. tarafından, Uygulama Arabirim Şartnamesi (AIS-Application Interface Specification) diye adlandırılan ve katı modelleme sistemleri için bir arabirim prosedürü geliştirildi. 1980'li yıllarda ise, CAM-I veri değişim problemini yeniden



gözden geçirdi ve Deneysel Sınır Dosya (XBF- Experimental Boundary File) diye adlandırılan yeni bir dosya şartnamesi hazırladı.



Bu arada, İlk Grafik Değişim Şartnamesi yani kısaca IGES (Initial Graphics Exchange Specification) denilen ve bilgisayarlı çizim sistemlerinin veri tabanları arasında veri değişimini sağlamaya yönelik bir başka dosya şartnamesi geliştirildi. Fakat o sıralar, IGES'in kapsam alanı, ayrıntılı bir çizimin gerek duyduğu tipik geometrik ve grafiksel gereksinimlerle sınırlıydı.

O zamandan bu yana ise, IGES, heterojen CAD sistemleri arasındaki verilerin değişimi için en çok kullanılan veri değişim standardı haline geldi. Bugün ise, IGES katı modelleme sistemleri arasındaki veri değişimlerini de kapsamaktadır.

### 3.6.1 SET (Değişim ve Transfer) Standardı

Fransadaki Aerospatiale firması tarafından geliştirilen bu veri değişim şartnamesi Standard d'Exchange et de Transfer veya kısaca SET diye anılmaktadır. Bu standardın geliştirilmesinin ardındaki gerçek ise IGES standardının kendi ihtiyaçlarını aşağıda açıklanan nedenlerden dolayı karşılayamamasıydı. Bu iki neden ise;

1. IGES dosya büyüklüğü çok fazlaydı, yani çok fazla yer kaplıyordu
2. Kendi CAD sistemlerine ait verilerinininlerinin tamamının IGES aracılığıyla transferi mümkün değildi.

Bu standard her ne kadar IGES veri değişim standardını baz almışsa da farklı bir dosya formatına sahiptir. Fakat bu format sayesinde, IGES ile kıyaslandığında dosya büyüklüğünde dikkate değer derecede bir azalma sağlanmıştır.



### 3.6.2 VDAFS Standardı

VDAFS (Verband der Deutschen Automobilindustrie - Flächenschnittstelle) denilen bu standart ise, IGES'in serbest formdaki yüzey verilerinin değişimi konusundaki yetersizliği sebebiyle, otomobil üreten Alman firmalarının ulusal, uluslararası, ve yan sanayi ile olan veri değişim işlemlerini sağlamak amacıyla geliştirilmiştir. Fakat VDAFS, CAD prosesinin sadece çok dar bir alanına hitap etmektedir.

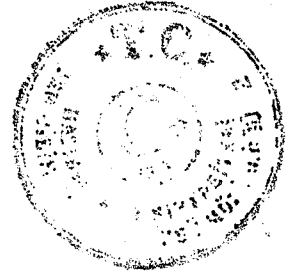
### 3.6.3 PDDI Standardı

PDDI (Product Data Definition Interface) ürün veri tanımlama arabirimi olarak çevirebileceğimiz bu standart ise, USA hava kuvvetleri tarafından tasarım ve imalat arasında tam bir bilgisayar tabanlı ürün tanımlama verisi arabirimi oluşturmak amacıyla tasarlanmıştır.

Projenin asıl amacı ise, parça kontrol bilgisi, parça özellikleri, toleranslar, B-Rep katıları, ve geometri arasındaki veri transferi için etkinliği arttırmaktır. Sonuç olarak da, veri yapıları bu ihtiyaçlara cevap verebilecek biçimde düzenlendi ve IGES'den daha etkili olacak bir dosya formatının içine yerleştirildi.

### 3.6.4 PDES Standardı

PDES (Product Data Exchange Standard) ürün veri değişim standardı da, 1983 yılında, yine IGES organizasyonu tarafından, 1980'li ve 1990'lı yılların ihtiyaçlarını karşılayabilmek üzere ne yapılabileceğinin belirlenmesi konusunda yapılan çalışmalar sırasında ortaya çıkmıştır. Bu standart IGES'in sahip olduğu zayıflıkları, ki bunlar dosya büyüklüğünün, ve bilgisayar işlem zamanının çok fazla olması, ve örneğin iş idaresi bilgisi, imalat proses bilgisi gibi, bilgi transferinden ziyade sadece veri transferi üzerinde yoğunlaşmış olması gibi bazı olumsuzlukları elimine etmek amacıyla düşünülmüştür[53].



### 3.6.5 ESPRIT Projesi 322 : CAD\*I

CAD\*I projesi ilk kez 1984 yılında, CAD veri değişimi için ve veri tabanı sistemleri ile sonlu elemanlar analizi sistemlerine arabirimler geliştirmek amacıyla başlatıldı [54].

ESPRIT projesi 322: CAD\*I'nın asıl amaçları ise aşağıda sıralandığı gibidir;

- ürün analiz verilerinin değişimini sağlamak üzere neutral formdaki dosya için bir spesifikasyonun geliştirilmesi,
- sonlu elemanlar analizi için ön ve son işlem birimlerinin geliştirilmesi,
- geliştirilen tekniklerin uygun test dosyalarının değişimi suretiyle test edilmesi,
- uluslararası standartlaştırma faaliyetlerine katkıda bulunmak [55].

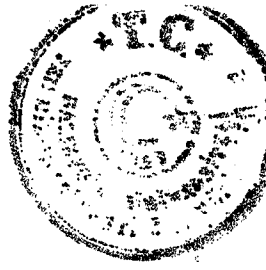
### 3.6.6 Uluslararası Standartlar Kurumu (ISO)

1983 yılında, 184 nolu ISO teknik komitesinin içinde tek bir uluslararası standart geliştirmek düşüncesiyle bir alt komite kuruldu. Bu proje için düşünülen başlık ise STEP (Standard for the Exchange of Product Model Data) idi. STEP veri değişim standardı PDES filozofisi ve tekniği üzerine kurulmuştur.

### 3.6.7 STEP standardının metodolojisi ve amacı

Standardın amacı bir endüstriyel ürünü tanımlamak için kullanılan bilginin değişimini kolaylaştırmaktır. Standart, farklı bilgisayar ortamları arasındaki iletişimi destekleyecek, ve tasarım ve analiz gibi çeşitli ürün yaşam süreci fonksiyonlarını yerine getiren sistemleri birbirlerine entegre etmeyi kolaylaştıracaktır [56].

STEP'in faaliyet alanı, ürünü yaşam süreci boyunca desteklemek ve tamlığını ve bütünlüğünü bozmaksızın neutral formdaki, bilgisayara yüklenmiş bir ürün modelini olduğu gibi kapsamaktır [54, 57].

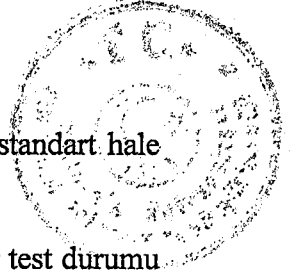


STEP standardının amaçlarını ise aşağıdaki gibi özetlemek mümkündür:

- bir ürünün tam bir tanımlamasını yapmak,
- heterojen sistemler arasında bilgi değişiminin herhangi bir yanlışlığa meydan vermeyecek yeni teknolojilerin uygulanmasını hızlandırmak.

STEP standardı diğer standartlara göre bazı önemli özelliklere sahiptir. Bunlar;

1. **Bütün ürün verisini kapsamı.** *Ürün verisi* bir ürünün her bir biriminin, başlangıçtaki gereksinimlerden onun ıskartaya çıkarıldığı andaki konfigürasyonuna kadar olanları içine alan yaşam süreci boyunca fiziksel karakteristiklerini ve işlevsel yapısını tanımlar. STEP standardı gerçekte bilgi değişimi için tasarlanmış olmasına rağmen değişim esnasında bilgi veri haline dönüştürülerek ifade edilir. Verilerin formu ise, transferin yapılacağı karşı taraftaki bilgisayarın değişim teknolojisine bağlıdır.
2. **Bilgi modelleme dilleri üzerine kurulması.** Bilgi modellemesi ne yapılabileceğini betimler ve sistem ihtiyaçlarını sadece bir kez belirleyip bunları, teknoloji ne kadar değişirse değişsin her seferinde geriye dönüp sıfırdan başlamak ve yeniden belirlemek yerine o anda uygun olan teknoloji neyse onunla örtüştürme imkanını sağlar. Bilgi modellemesi, bilginin veriler aracılığıyla nasıl ifade edilebileceği veya bilginin bir bilgisayar sistemi tarafından nasıl manipüle edilebileceği konusuyla ise uğraşmaz. O sadece bilginin ifade edilmesiyle ve özelliğini bozmadan aynen elde etme ile ilgilidir.
3. **Bilgi modelini verilerle örneklenmiş modelden ayırması.** STEP, Express denilen bilgisayar programlama dili kullanılarak tanımlanmış ve ifade edilmiştir. Express programlama dili, bilgi modellerinin bilgisayarlar tarafından anlaşılıp, kullanılabilir bir formda ifade edilmesi için tasarlanmıştır. STEP Express modeli ise, varlıkları, STEP uygulamaları tarafından kullanılacak şekilde belirler ve tanımlar.
4. **Uygulama protokollerine göre gerçekleştirilmesi.** *Uygulama Protokolü* (application protocol), özel bir uygulama için bilgi gereksinimleri ve konu olan alana uygun entegre edilmiş kaynakların kullanımını tarif eder. Uygulama protokolleri, içeriği, kullanımı, ve STEP'in mutlaka içermesi gereken ve bir ürünün yaşam sürecinde yer alan analiz gibi, özel bir amaca yönelik bir çeşit ürün verisini tanımlayan standartlardır. Uygulama protokolleri, belirli işlemleri güvenilir bir



biçimde ve sürekli olarak desteklemek üzere STEP'in kullanılmasını standart hale getirirler.

5. **Uyumluluk testine gereksinim duyması.** Uyumluluk testi, belirli bir test durumu için standardın farklı biçimde uygulanmasını önlemek amacıyla düşünülmüştür. STEP'in uygulama testinin amaçları ise ;

- (a) **Tekrarlanabilirlik :** Test sonuçları, ne zaman yapılırsa yapılsın hep aynı ve tutarlıdır.
- (b) **Karşılaştırılabilirlik :** Test sonuçları ne zaman elde edilirse edilsin birbirleriyle uyumludur.
- (c) **Denetlenebilirlik :** Test işlemleri, doğru yapılmaları durumunda, elde edilen test sonuçlarının gözden geçirilmesiyle denetlenebilir ve sağlanması yapılabilir.

### 3.6.8 STEP Dokümanları

STEP standardı, amaca göre sınıflara ayrılan ve bölüm diye adlandırılan bir dizi dokümandan oluşturulmuştur. Esas olarak, STEP dokümantasyonu yedi ana kategoriye ayrılabilir. Bunlar Şekil 3.15'de grafik olarak verilmiştir.

#### 1. Genel bakış :

Bu STEP'in genel giriş bölümüdür ve standardın bu bölümüne bölüm numarası olarak "1" verilmiştir.

#### 2. Tanımlama Metodları :

Ürün tanımlaması için gerekli olan bilgileri belirtmek için kullanılan EXPRESS bilgi modelleme dilinin özelliklerini içermektedir. Bu bölüm için verilen bölüm numaraları 11, 12 diye devam eder.



### 3. Uygulama Metodları :

STEP ürün bilgilerinin, bilgi değişim ortamında fiziksel olarak nasıl ifade edileceğine dair özellikleri sağlar. Bu bölüm için verilen bölüm numaraları ise 21, 22 diye gitmektedir.

TANIMLAMA		UYGULAMA PROTOKOLLERİ				UYGUNLUK VE ARAÇLAR	
		#201 Detaylı çizim	#202	#205 #203	#204		
#11 EXPRESS Dili Uygulama Formu		<b>TÜMLEŞİK KAYNAKLAR</b> <b>Uygulama Kaynakları</b>				#31 Test Etme Konsepti	
#12 EXPRESS-I		#101 Teknik resim çizimi	#104 FEM				
		<b>Genel Kaynaklar</b>					
		#46 Sunum	#44 Konfigürasyon				
		#48 Form Nitelikleri	#45 Malzeme				
		#42 Şekil Sunumu					
		#41 Tanımlama ve Destek					
		#21 Fiziksel Dosya	#22 Kullanılabilir form	Veri Tabanı	Bilgi Tabanı		
		<b>UYGULAMA METODLARI</b>					

Şekil 3.15 STEP dokümantasyonunun mimarisi [53]



### 1. Uygunluk ve Araçlar :

STEP standardının uygunluk testi için özellikleri içerir. Testin amacı, bir ürünün standart tarafından istenen belirli özelliklerin varlığını ve uygunluğunu belirlemektir. Bölüm numarası olarak ise 31, 32 diye devam eden seri verilmektedir.

### 2. Kaynak Bilgi Modeli (Genel Kaynaklar) :

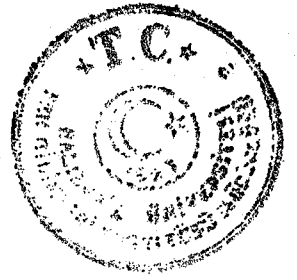
Bu bölüm, şekil betimlemeleri, malzemeler, form özellikleri, toleranslar v.s. gibi farklı türdeki uygulamaları destekleyen bilgi modellerinin özelliklerini içermektedir. Bölüm numarası olarak ise 41 ve daha yukarısı verilmektedir.

### 3. Uygulama Bilgi Modelleri (Uygulama Kaynakları)

Bu bölüm ise, belli bir uygulama alanı için yapılmış olan bilgi modelleri ile ilgilidir. Bu uygulamalar, çizim, sonlu elemanlar yöntemi ile analiz v.s. gibi uygulamalar olabilir. Bu bölüm de, 101 ve daha yukarısı ile numaralandırılır.

### 4. Uygulama Protokolleri :

Bir uygulama protokolü faaliyet alanını ve spesifik bir uygulama için gerekli bilgileri belirler. Uygulama protokolleri (AP's - application protocols ) STEP standardının ana bölümlerini teşkil etmektedir. Uygulama protokolleri, yüzey temsili yöntemi ile mekaniksel bir tasarımın yapılması gibi spesifik bir uygulama alanı için bilgi değişimi gereksinimlerini desteklemek üzere yapılırlar (Uygulama protokolu Bölüm 205)[58]. Uygulama protokolleri özellikle spesifik uygulama sistemleri (örneğin 2 farklı sonlu elemanlar analiz sistemi) arasında veri değişimi için gerekli olan STEP bilgi modelinin alt kümelerini belirtmektedir.

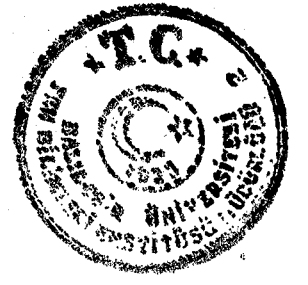


### 3.7 Bölümsel Sonuç ve Değerlendirmeler

Bu bölümde, bilgisayar destekli mühendislik faaliyetlerinin bütünleştirilmesine gerek olduğu anlaşılana kadar izlenen yöntemler incelenmiş, bu faaliyetlerin entegrasyonu için gerekli olduğu düşünülen Ürün Modelleme kavramı ve bu aşamanın bir önceki aşaması olan Geometrik Modelleme konusuna değinilmiş, Leeds üniversitesinde bu konuda yapılan çalışmaya detaylı olarak yer verilmiş ve ardından da diğer araştırmacıların konu ile ilgili yaptıkları çalışmalar gözden geçirilmiştir.

Ayrıca, halen günümüzde de kullanılmakta olan ve Ürün Modelleme kavramının hayata geçirilmesi durumunda kullanılmasına gerek kalmayacağı düşünülen veri değişim standartları da değerlendirilmiştir. Öncelikle, günümüze kadar ortaya atılan veri değişim protokolleri tarih sırasına göre incelenmiş ve hali hazırda kullanılmaya başlanan ve bütün bu protokollerin yerine geçecek olan STEP standardına da yer verilmiştir.





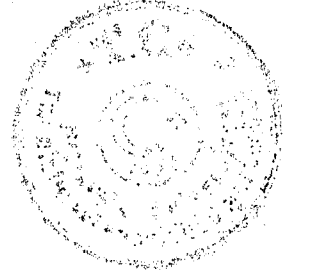
## 4. BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMIĞI (CFD) İÇİN BİR AKTİVİTE ANALİZİ

### 4.1 GİRİŞ

Bu bölümde, SADT (Structured Analysis and Design Technique) denilen ve aktivitelerin modellenmesinde kullanılan bir metod yardımıyla bilgisayar destekli akışkanlar dinamiği analizinde yer alan aktiviteler incelenecektir. Ayrıca bilgisayar destekli akışkanlar dinamiği analizi için yaratılan modelin önemli adımları da yine bu bölümde tartışılacaktır. Bu çalışmanın önemli katkılarından biri de, bu bölümde verilecek olan, bilgisayar destekli akışkanlar dinamiği analizindeki aktivitelerin bir modelleme tekniği kullanılarak analiz edilmesidir.

Bu çalışmada yapılan araştırmada, aynı zamanda bilgisayar destekli akışkanlar dinamiği analizini destekleyecek bir ürün veri modelinin oluşturulması da amaçlanmaktadır. Fakat böyle bir ürün veri modeli geliştirilmeden önce bilgisayar destekli akışkanlar dinamiği analizinde yer alan aktivitelerin incelenmesi ve onların SADT denilen modelleme tekniği ile modellenmesi hem bilgisayar destekli akışkanlar dinamiği prosesinin daha iyi anlaşılması ve daha detaylı incelenmesi ve hem de mühendislik tasarım ve analiz aşamalarının entegrasyonunu kolaylaştıracak bir bilgisayar yazılımının geliştirilmesi açısından oldukça önemlidir.

Bilgisayar destekli akışkanlar dinamiği analizinin anlaşılması ve araştırılmasında kullanılacak olan SADT (Structured Analysis and Design Technique) modelleme yönteminin [59] kısa bir özeti bölüm 4.2 de verilecektir. Konu ile ilgili daha detaylı açıklamalar Ek-A da bulunabilir. İlerleyen bölümlerde ise, tüm bilgisayar destekli akışkanlar dinamiği analizi prosesi boyunca yer alan aktiviteler tanımlanacaktır.

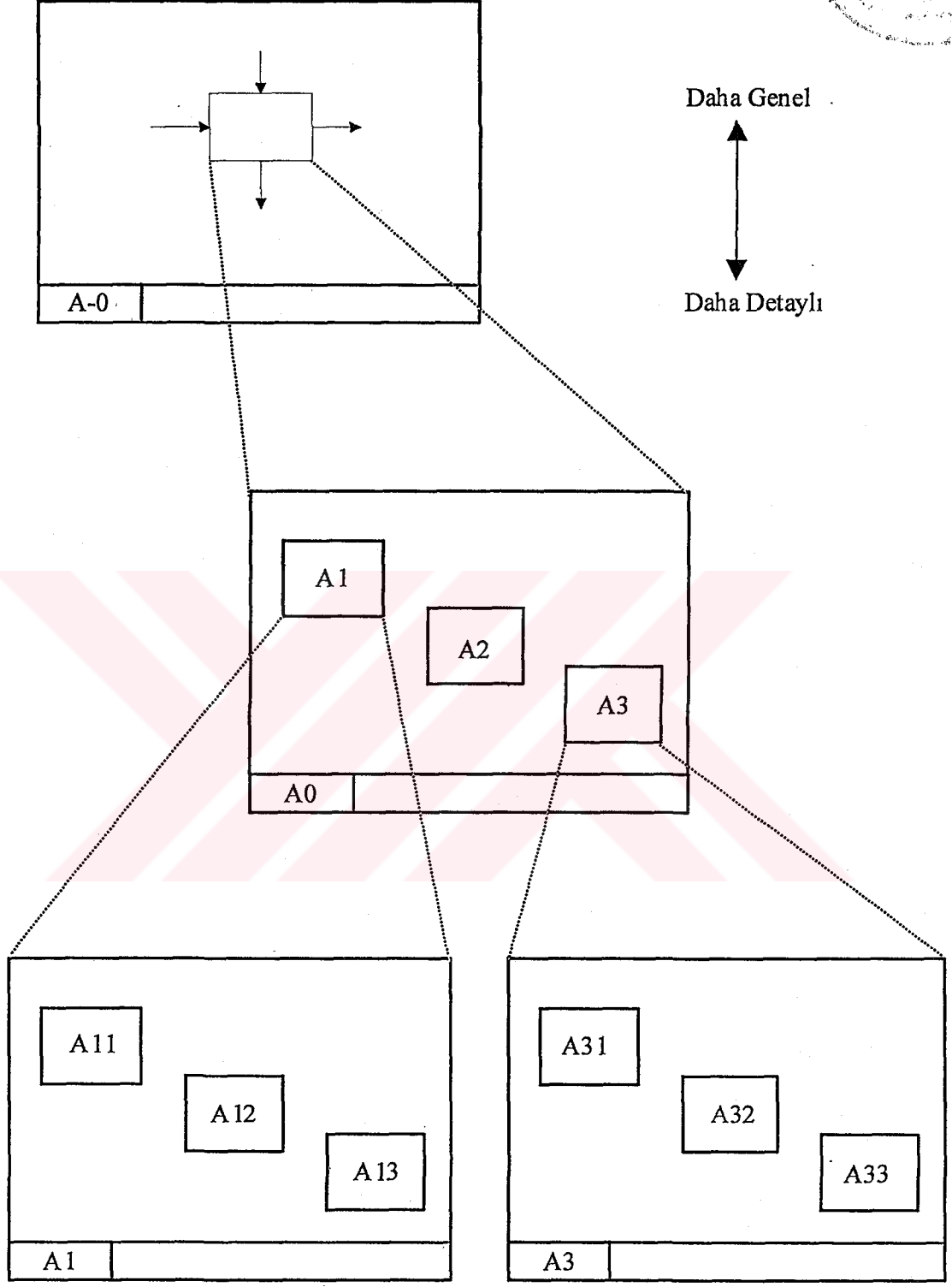
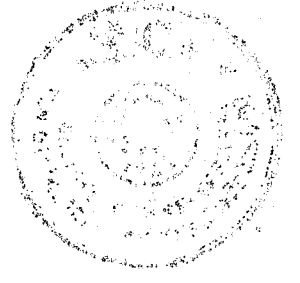


## 4.2 SADT (Structured Analysis and Design Technique)

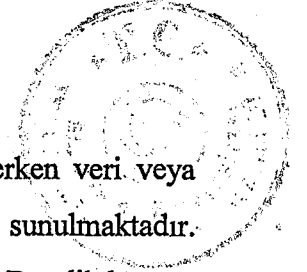
Yapısal Analiz ve Tasarım Tekniği (SADT) denilen bu teknik SofTech firmasında çalışan Douglas T. Ross tarafından [60, 61, 62, 63] geliştirilmiş ve Amerikan hava kuvvetleri bilgisayar destekli imalat projesinde ve yine Amerikan hava kuvvetleri bütünleştirilmiş bilgisayar destekli imalat projesi kısaca ICAM projesinde[64] kullanılmıştır. ICAM projesi süresince de, IDEF0 (ICAM DEFinition Language Zero) denilen ve yukarıda adı geçen tekniğin bilgisayar aracılığıyla kullanılmasını sağlayacak bir bilgisayar yazılımı geliştirildi. IDEF0 yazılımı, bir prosesdeki aktivitelerin grafiksel bir biçimde gösterilmesini sağlayan ve SADT tekniğinin bir alt kümesi olan bir bilgisayar yazılımı olarak da tanımlanabilir. IDEF0, bir sistemdeki aktivitelerin ve bu aktiviteler arasındaki ilişkilerin girdiler, çıktılar, kontroller ve aktivitelerin yapılmasını sağlayan mekanizmalar aracılığıyla ifade edilmesini sağlamaktadır [65].

Bir SADT modelindeki ilk diyagram, tüm sistemin genel bir tanımını içermekte ve ilk aşamadaki bu diyagrama *konteks diyagramı* denilmektedir. Bu diyagramda, bir prosesin her bir ana unsuru küçük dikdörtgenlerle ifade edilmektedir. Bu dikdörtgenlerin daha detaylı halleri ise yine benzer biçimde diğer diyagramlarla gösterilmektedir. Her dikdörtgen, istenilen seviyedeki detaya ulaşıncaya kadar daha alt dikdörtgenlere ayrılabilir. Hiyerarşik olarak bir üst seviyede bulunan dikdörtgen kendine göre daha alt seviyede bulunan bir dikdörtgene göre *ebeveyn* durumundadır.

En üst seviyede yer alan diyagram, sadece tek bir dikdörtgen içermekte ve A-0 olarak numaralandırılmaktadır. Bu konteks diyagramının ayrıştırılmış biçimi birden fazla dikdörtgen ve ilişki okları içermekte ve A0 olarak indekslenmektedir. Bu A0 diyagramı içerisindeki dikdörtgenler ise, sağ alt köşelerine A1'den A6'ya kadar numara verilerek indekslenir. Eğer gerekliyse bu dikdörtgenler de Şekil 4.1 de görüldüğü gibi ayrıştırılabilir. Dikdörtgenlerin indekslenmesinde kullanılan A harfi ise aktivitenin baş harfini simgelemektedir. Bir SADT modelinde, en üst seviyede yer alan diyagramlar üstü kapalı olmasına rağmen tüm bilgiyi içerirken en alt seviyedeki diyagramlar ise daha az bilgi fakat daha fazla detay içermektedirler [66].



Şekil 4.1 SADT diyagramlarının hiyerarşik yapısı



Her bir SADT diyagramında aktiviteler dikdörtgenlerle ifade edilirlerken veri veya bilgiler ve aktiviteler arasındaki ilişkiler ucu oklu çizgiler aracılığıyla sunulmaktadır. Bir SADT diyagramında altıdan fazla dikdörtgen kutu bulunamaz. Bu dikdörtgen kutulara giren oklar o aktiviteyi yapmak için gerekli verileri ifade ederken dışarıya doğru çıkan oklar ise o aktivite tarafından üretilen verileri göstermektedir. Bir SADT diyagramında dört değişik veri türü mevcuttur. Bunlar ;

- **Girdi** : Bir aktiviteyi gerçekleştirmek için gerekli veriyi ifade eder. Bu girdi, aktivite tarafından kullanılacak ve çıktıya dönüştürülecektir. Girdi oku, aktivite dikdörtgenine sol taraftan girmekte ve  $I_n$  olarak endekslenmektedir.
- **Çıktı** : Aktivite tarafından yaratılan veriyi göstermektedir. Aktivite dikdörtgeninin sağ tarafından çıkmakta ve  $O_n$  olarak endekslenmektedir.
- **Kontrol** : Bir aktiviteyi etkileyecek bilgiyi tanımlamak üzere kullanılır. Her aktivite dikdörtgeni en az bir kontrol okuna sahiptir. Kontrol bilgisi, aktivite dikdörtgenine üst kenardan girer ve  $C_n$  olarak endekslenir.
- **Mekanizma** : Aktiviteyi yürüten bir cihazı, kişiyi veya aracı temsil etmektedir. Bu, aktivite dikdörtgenine alt kenarından girer ve  $M_n$  olarak endekslenir.

SADT tekniği ile ilgili detaylı bilgi Ek-A da verilmiştir. Konuyla ilgili çok daha detaylı bilgi ise [67] nolu kaynaktan temin edilebilir.

#### 4.3 CFD'de Yer Alan Aktivitelerin Analizi

Bu bölümde, bilgisayarlı akışkanlar dinamiği analizinde CFD (Computational Fluid Dynamics) yer alan tüm aktivitelerin SADT tekniği kullanılarak modellenmesinden ve yine bu aktivitelerin detaylarına ayrıştırılmasından bahsedilecektir.



#### 4.3.1 Konteks Diyagramı (A-0)

Şekil 4.2’de, CFD analiz prosesinin SADT tekniği kullanılarak yapılmış modelinin konteks diyagramı (A-0) görülmektedir. Bu diyagram, modelleneyecek olan aktiviteyi genel olarak göstermektedir. Bu aktivitenin ayrıştırılması ise prosesin içerdiği aktivitelerin incelenmesi amacına yöneliktir.

CFD analizine giren girdiler ise, hız ve sıcaklık değerleri gibi akışkana uygulanabilecek sınır ve başlangıç koşullarını tanımlayan *CFD Sınır ve Başlangıç Koşulları*, analizi yapılacak bilgisayar destekli akışkanlar dinamiği alanıyla ilgili geometrik verileri ifade eden *CFD Alan Geometrisi*, akışkanın, fiziksel (vizkozite, yoğunluk) ve termal özellikleri gibi özelliklerini tanımlayan *Akışkan Özellikleri*’dir. Çıktı ise, hız, basınç değerleri gibi analiz sonuçları değerlerini yansıtan *Hesaplanmış Özellik Değerleri*’dir.

Şekil 4.2’den de görüleceği üzere analiz esnasında kullanılan 6 adet mekanizma vardır. Bunlar, *Mühendislik yargısı*, *Tasarım gereksinimleri*, ve *Prensipier ve Sınırlamalar*’ın kontrolü altında proseste önemli rol oynayan *Analist* (kullanıcı veya tasarımcı da denilebilir), ısı transferi analizi, bilgisayar destekli akışkanlar dinamiği analizi gibi analizin doğal davranışını sınıflandıran *Fiziksel Olay*, fiziksel olayı modelleyecek teorinin belirlenmesinde kullanılan *Fiziksel Teoriler*, modelin kolayca çözümlenmesinin kolaylaştırılmasını sağlayacak deneyime ya da sezgiye dayalı bir takım kararları ifade eden *Kabuller* ve mevcut olan matematiksel denklemleri, çözüm tekniklerini ve stratejilerini ifade eden *Çözüm Teknikleri ve Stratejiler* ve son olarak CFD analizine destek olacak FIDAP (Fluid Dynamics Analysis Package) [80] gibi hali hazırda mevcut yazılım paketlerini temsil eden *Araçlar*’dır.



#### 4.3.2 Konteks Diyagramının (A0) Ayrıştırılması

İlk diyagram, Şekil 4.3'den de görüleceği üzere altı farklı alt aktiviteye ayrılmıştır. Diyagramdaki ilk aktivite "*Fiziksel Modelin Oluşturulması*" diye adlandırılan ve matematiksel modelin çok daha genel bir biçimi olan, fiziksel teoriler, kabuller ve fiziksel olay hakkında bilgiler içeren fakat sadece belirli bir problem için özelleştirilmemiş olan fiziksel modelin oluşturulmasını sağlamaktadır. Analiz prosesindeki ikinci aktivite ise, modellenecek belirli bir probleme uyacak denklem takımlarının oluşturulduğu, CFD sınır ve başlangıç koşulları, CFD alan geometrisi gibi bazı ürün bilgilerinin idealleştirildiği "*Matematiksel Modelin Oluşturulması*"dır. "*Analiz*" aşamasında, matematiksel model sonlu elemanlara ayrılır ve denklem takımları genellikle FIDAP gibi nümerik analiz paketi yardımıyla çözülür. Daha sonra analiz sonuçları, sonuçların bilgisayar ortamında görüntülenmesi amacıyla kullanılır ve ardında da analizi yapan kişi tarafından "*İdealleştirme Hatalarının Kontrolü*" aşamasında değerlendirilir. Eğer çözüm sonucu elde edilen değerler memnuniyet verici bulunmazsa, değerlendirme sonucu varılan yargılar "*Matematiksel Modelin Oluşturulması*" aşamasını kontrol eden idealleştirmelerin değiştirilmesinde kullanılırlar.

#### 4.3.3 Fiziksel Modelin Oluşturulması (A1)

Şekil 4.4, bu aktivitenin ayrıştırılmış halini göstermektedir. Diyagramdaki ilk aktivite olan "*Uygun Olayın Seçilmesi*" aşamasında, analizi yapan kişi tarafından fiziksel bir olayı en uygun biçimde tanımlayacak uygun bir fenomen seçilir. Seçilen bu fenomen analiz prosesi boyunca hesaplanacak olan özellik değerleri üzerine etki edebilecek bir potansiyele sahip olmalıdır.

Diyagramdaki ikinci aktivite ise A12 olarak indekslenen "*Fiziksel Teorilerin Seçilmesi*"dir. Uygun bir fenomenin seçilmesine müteakiben, fiziksel fenomeni modellemek için bir fiziksel teorinin de seçilmesi gerekmektedir. Bazı durumlarda fiziksel bir olay için birden fazla uygun fiziksel teori bulunabilir. Bu durumda ise yapılacak olan, en uygun olanın seçilmesidir. Örneğin, ısı iletimi, ısı taşınımı ve ısı radyasyonu, bir fiziksel



olayın modellenmesi için uygun olabilirse de, olay üzerinde diğerlerine nazaran daha fazla etkili olabilecek olan bir fiziksel teorinin seçilmesi daha düşük maliyetli olmasını sağlayacaktır.

Bu aşamadaki üçüncü ve son aktivite ise A13 ile indekslenen ve “*İlave Kabullerin Uygulanması*” diye adlandırılan aktivitedir. Bazı durumlarda, problemi daha da basit hale getirecek fakat teorinin özünü değiştirmeyecek, örneğin akışkanın vizkoz olmaması veya vizkozitenin Newtonyen olması veya akışkanın sıkıştırılmaz olması gibi, bir takım ilave kabullerin yapılması mümkün olabilir.

#### **4.3.3.1 Uygun Fenomenin Seçilmesi (A11)**

Adı geçen faaliyet sırasında, C1, C2, C3, C4 diye indekslenen kontroller göz önüne alınarak uygun fenomenin tanımlanması söz konusudur. Bu aktivitenin daha alt aktivitelere ayrılmış biçimi Şekil 4.5’de gösterilmiş olup, bu alt aktivitelere ait kısa tanımlamalar ise aşağıda sırasıyla verilmiştir.

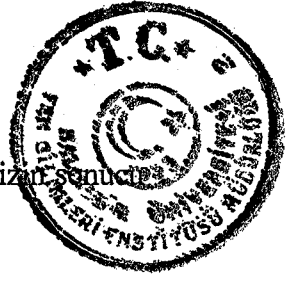
##### **4.3.3.1.1 Aday Fenomenin Belirlenmesi (A111)**

Bu aktivitede, talep edilen özellik değerlerini etkileyebilecek fiziksel bir olayı ya da olayların analizini yapan kişi tarafından belirlenir. Fiziksel olayların listesi, bu aktiviteye girdi verisi olarak girerken, aktivitenin devamı esnasında da aday fenomen belirlenir ve Şekil 4.5’den de görüleceği gibi “*Aday Fenomen*” aktiviteyi çıktı olarak terk eder.

##### **4.3.3.1.2 İlave Özelliklerin Belirlenmesi (A112)**

Eğer, talep edilen özellik değerlerini etkileyebilecek nitelikte olan seçilen aday fenomen, diğer başka fiziksel özelliklerden etkileniyorsa, bunlar bu aktivitenin





içerisinde belirlenirler. Diğer fiziksel özelliklerin kaynağı ise bir başka analizdir. Sonuçları da olabilir.

#### 4.3.3.1.3 Ortama Uygunluğun Değerlendirilmesi (A113)

Bu aktivitede, seçilen aday fenomen, analizin sonuçları üzerindeki etkisi dikkate alınarak yeniden değerlendirilir. Eğer belirlenmiş olan aday fenomenin, özellik değerleri üzerindeki etkisinin küçük olması nedeniyle uygun olmadığına karar verilmesi durumunda bu aday fenomen elenir. Bu aktivitenin çıktısı ise “*Uygun Fenomen*” dir.

#### 4.3.4 Matematiksel Modelin Oluşturulması Aktivitesinin Ayrıştırılması (A2)

CFD analizinde önemli bir yere sahip olan matematiksel modelin oluşturulması aktivitesi, Şekil 4.6’da olduğu gibi daha alt aktivitelere ayrıştırılmıştır. Sözü edilen şekildeki diyagramdan da görüleceği gibi, bu aktivitede esas olarak iki alt aktivite vardır. Bunlar sırasıyla “*CFD problemi bilgilerinin idealleştirilmesi*” ve “*Denklemlerin formüle edilmesi*”. Bu aktivitelere ait bilgiler ve yine bu aktivitelere ait içerisinden yer alan alt aktiviteler ile ilgili bilgi ve açıklamalar aşağıda verilmiştir.

##### 4.3.4.1 CFD Problem Bilgilerinin İdealleştirilmesi (A21)

Bu aşamada yer alan alt aktiviteler, Şekil 4.7’deki diyagramda gösterilmiştir. İdealleştirme prosesinde, aslında analiz için gerekli olmayan bir takım bilgilerin çıkarılması veya kabuller halinde bazı bilgilerin ilave edilmesi işlemleri yapılır. Probleme ait bilgilerinin idealleştirilmesi, karmaşıklığı azaltmak suretiyle matematiksel modelin basitleştirilmesini sağlamaktadır. Örneğin boyutların idealleştirilmesi, problemi iki boyutlu ve hatta bazen de bir boyutlu hale kadar indirgeyebilir. Geometrik idealleştirmelerde ise, önemli olmayacağı düşünülen bazı





geometrik detayların kaldırılması söz konusudur. Bu aşamada yeralan dört alt aktivite aşağıda, diyagramdaki sıraya göre kısaca incelenecektir [68,69, 70, 71].

#### 4.3.4.1.1 Boyutların İdealleştirilmesi (A211)

Bu aktivitede, problemin boyutu mümkün olması durumunda üç boyutludan iki boyutluya veya iki boyutludan bir boyutlu hale getirilir. Bu boyut azaltma işlemi, analizi yapacak olan kişi tarafından, matematiksel modelin sistem denklemlerindeki bir veya iki boyutun elimine edilerek basitleştirilmesiyle yapılır.

#### 4.3.4.1.2 Akışkan Özelliklerinin İdealleştirilmesi (A212)

Bu aktivite sırasında, akışkan özellikleri ya deneysel bulgular ya da önceki deneyimler göz önüne alınmak suretiyle idealleştirilir. Genel olarak söylemek gerekirse, bir mühendislik analizinde, akışkan sıkıştırılmaz ve Newtonyen kabul edilir. Bu idealleştirmeler de matematiksel modelin basitleşmesine neden olmaktadır.

#### 4.3.4.1.3 CFD Alan Geometrisinin İdealleştirilmesi (A213)

Bu aktivite süresince analiz edilecek alanın geometrisi, *mühendislik yargısı* ve bir önceki analiz sonucu elde edilen *idealleştirme değişiklik talebi* kontrolleri dikkate alınarak, analizi yapan kişi tarafından idealleştirilir. Bir çok mühendislik analizinde, analiz edilecek olan geometri daha çabuk ve işe yarar sonuçlar elde etmek amacıyla basitleştirilmektedir. Fakat bu durumda elde edilecek sonuçlar ancak belirli bir dereceye kadar doğru olacaktır. Mühendislik analizinde en çok görülen geometrik idealleştirme ise yuvarlatılmış köşelerin keskin köşeler olarak değiştirilmesidir [72]. Detayları ihmal ederek gerçekleştirilen geometrik idealleştirmeler, talep edilen özellik değerlerini dikkate değer derecede etkilemeyecek fakat gerekli olan bilgisayar çabasını



da önemli ölçüde azaltacaktır.

#### 4.3.4.1.4 Sınır ve Başlangıç Koşullarının İdealleştirilmesi (A214)

Aktivite boyunca mümkün olması durumunda sınır şartları ve başlangıç koşulları üzerine kabuller uygulanır. Bazı durumlarda sınır ve başlangıç koşulları değerlerini tam olarak koymak mümkün olmayabilir. Örneğin, doğaya özgü sınır koşulları bunların ihtimale dayalı tabiatlarından dolayı hiç bir zaman tam olarak belirlenip yerine konamaz.

#### 4.3.4.2 Denklemlerin Formülasyonu (A22)

Aktivite süresince, fiziksel teori ve talep edilen özellik değerleri arasındaki ilişki spesifik denklemler aracılığıyla belirlenir, matematiksel denklemler boyutsuz hale getirilir ve matematiksel denklemleri çözmek üzere çözüm teknikleri ve stratejileri seçilir. "*Denklemlerin Formülasyonu*" aktivitesinde yer alan üç alt aktivite ise Şekil 4.8'de verilmiş ve bu alt aktiviteler diyagramda yer aldıkları sıraya göre aşağıda kısaca açıklanacaktır.

##### 4.3.4.2.1 Temel Denklemlerin Çıkarılması (A221)

Bir fiziksel model boyutlara bağlı olmayan bir formda olup, talep edilen özellik değerleri ve fiziksel teoriler arasındaki ilişkileri, matematiksel denklem takımları aracılığıyla bulmak üzere bir matematiksel modele dönüştürülmelidir. Fiziksel bir teorinin tanımına bağlı olarak matematiksel denklemlerin kurulması bu aktivite içerisinde yer almaktadır. Bu aktivitenin sonucunda elde edilen çıktı ise "*Temel Denklemler*" dir.



#### 4.3.4.2.2 Sistem Denklemlerinin Boyutsuz Hale Getirilmesi (A222)

Bu aktivitede sistem denklemleri analizi yapan kişi tarafından "İdealleştirilmiş CFD problemi bilgileri" nin kontrolü altında boyutsuz hale dönüştürülmektedir. Bir problemin, karakteristik hız, karakteristik uzunluk gibi karakteristik değerlerle boyutsuz hale getirilmesi, aşağıda maddeler halinde verilen bir takım yararlar sağlar :

- Problemin zorluk derecesinin tahmin edilmesi. Örneğin Re sayısının tahmin edilmesi buna bir örnektir. Eğer momentumun korunumu denklemini dikkate alacak olursak;

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = -\frac{\partial P}{\partial x} + \mu \left[ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right] \quad (4.1)$$

ve bu denklemi karakteristik uzunluk  $L$  ve karakteristik hız  $U_\infty$  yi kullanarak boyutsuz hale getirecek olursak aşağıdaki denklemi elde ederiz.

$$\rho \frac{U_\infty^2}{L} + \rho \frac{U_\infty^2}{L} = \frac{U_\infty^2}{L} + \mu \frac{U_\infty}{L^2} \quad (4.2)$$

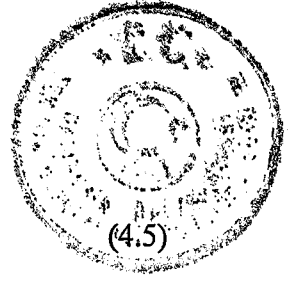
Eğer bu denklem aşağıda verilen terim ile bölünecek olursa

$$\rho \frac{U_\infty^2}{L} \quad (4.3)$$

o takdirde aşağıdaki sonuca varırız.

$$1+1=1+\frac{\mu}{\rho U_\infty L} \quad (4.4)$$

Buradan da görüleceği üzere denklemin son kısmı  $1/Re$  sayısına eşittir.



$$\frac{\mu}{\rho U_{\infty} L} = \frac{1}{Re} \quad (4.5)$$

Buradan,  $Re$  sayısı tahmin edilebilir ve problem hakkında bir fikir sahibi olunabilir. Örneğin  $Re$  sayısının değerinin 2000'den aşağı olması durumunda problemin bir laminar akış problemi olduğunun tesbit edilmesi gibi.

- Denklemdaki terimlerin bağıl olarak önemlerinin ölçülmesi. Örneğin 4.1'deki momentumun korunumunu ifade eden denklemin son kısmını dikkate alacak olursak;

$$\mu \left[ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right] \quad (4.6)$$

eğer akışkanın viskozitesi düşük ise, yukarıdaki terimin çok küçük bir değere haiz olacağını ve bu yüzden de bu terimin ihmal edilebileceğini söylemek mümkündür.

- Denklemdaki terimler arasında oluşabilecek büyüklük farklılıklarının azaltılmasına yardım eder.

#### 4.3.4.2.3 Çözüm Teknikleri ve Stratejilerinin Seçimi (A223)

Matematiksel denklemlerin kurulmasından sonra, denklemlerin çözümü için Sonlu Elemanlar gibi uygun bir çözüm tekniği seçilmeli ve çözüm stratejileri belirlenmelidir. Örneğin Quasi-Newton, Newton-Raphson metodu gibi.

#### 4.3.5 Analiz Aşaması (A3)

Bu aşama, şekil 4.3'deki diyagramda üçüncü aktivite olarak gösterilen ve CFD



analizinin yapıldığı bir aşamadır. Bu aktivitenin içerisinde yer alan diğer alt aktiviteler ise Şekil 4.9'da verilen diyagramda gösterilmiştir. Birbirini sırayla takip eden bu üç aktivitedeki "Matematiksel Modelin Diskretizasyonu" diye adlandırılan birinci ve "Diskretizasyon Hatalarının Kontrolü" denilen ikinci aktivite arasında tek bir geri besleme çevrimi mevcuttur. Bu çevrim, diskretizasyonun gerektiğinde yeniden iyileştirilmesine yardımcı olmaktadır. Matematiksel model oluşturulduktan sonra ise bu modelin diskretizasyonu yapılır ve ardından da çoğunlukla bir nümerik çözüm programı ile çözüme gidilir. Aşağıda bu aşamada yer alan alt aktiviteler sırasıyla tanımlanmış ve kısaca açıklanmıştır.

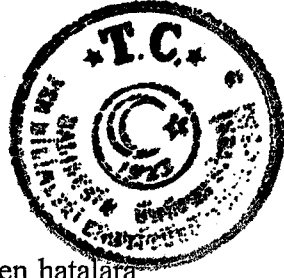
#### **4.3.5.1 Matematiksel Modelin Diskretizasyonu (A31)**

Eğer oluşturulan matematiksel modelin analitik olarak çözümü mümkün değilse bu durumda bilgisayar yaklaşımı bir çözüm yolu kullanılmalıdır. Bilgisayarlı yaklaşımı çözüm yolu ise tamamen sürekli problemin diskretizasyonu temeline dayalıdır. Bu aktiviteye girdi olarak giren matematiksel model, bu aktivite sırasında diskretize edilmekte ve sonuçta "diskretize edilmiş model" çıktı olarak aktiviteyi terk etmektedir.

Diskretize etme işlemi ile genellikle sürekli bir alanın belirli ebattaki ve sonlu sayıdaki elemanlara ayrılması ve de düğüm noktalarındaki fiziksel özelliklerin ifade edilmesi kastedilmektedir.

#### **4.3.5.2 Denklemlerin Çözülmesi (A32)**

Bu aktivitede, diskretize edilmiş matematiksel modele ait denklemler nümeriksel bir yöntem, örneğin Sonlu Elemanlar gibi, kullanılmak suretiyle çözülür ve ilk analiz sonuçları elde edilir.



#### 4.3.5.3 Elemanlara Ayırma Hatalarının Kontrol Edilmesi (A33)

İlk analiz sonuçları, bu aktivitede elemanlara ayırma sırasında meydana gelen hatalara karşı kontrol edilir, eğer yapılan bir hata mevcut ve bu yapılan hata kabul edilemeyecek bir seviyede ise o durumda “*diskretizasyon düzeltme talebi*” diye adlandırılan bir çıktı üretilir ve bu çıktı A31 olarak indekslenmiş bulunan ve yine analiz aktivitesi içerisinde birinci aktivite olan “*Matematiksel Modelin Diskretizasyonu*” adlı aktiviteye kontrol girdisi olarak gönderilir.

Eğer, matematiksel denklemlerin çözümü için herhangi bir nümerik çözüm tekniği kullanılacaksa bu durumda problemin diskretize edilmesi kaçınılmaz olmakta ve bu da sistemde bir takım diskretizasyon hatalarının oluşmasına yol açmaktadır.

#### 4.3.6 Sonuçların Bilgisayar Ortamında İşlenmesi (A4)

Bu aktivitenin amacı, analiz sonuçlarının değerlendirilebilmesine yönelik olarak elde edilen sonuçlardan yararlanarak diğer bazı fiziksel özelliklerin hesaplanması ve bunların bilgisayar ortamında görüntülenmesi veya canlandırılmasıdır. Herhangi bir şekilde işlenmemiş olan ham analiz sonuçları, bu aktiviteye girdi olarak girmekte ve daha sonra da bir yazılım paketi aracılığıyla analizi yapan tarafından bilgisayarda görüntülenmekte ve sonuçlar değerlendirilmeye çalışılmaktadır.

#### 4.3.7 İdealleştirme Hatalarının Kontrolü (A5)

Şekil 4.3’deki diyagramda yer alan son aktivite ise “*İdealleştirme Hatalarının Kontrolü*” dür. Yapılmış olan idealleştirmelerin geçerliliği, sonuçların, elde edileceği düşünülen sonuçlarla karşılaştırılmak suretiyle gözden geçirilir ve eğer gerekli ise “*İdealleştirme Düzeltme Talebi*” geri beslemesi üretilip, Şekil 4.3’deki diyagramda üçüncü aktivite olarak yer alan ve A3 olarak indekslenmiş olan “*Matematiksel Modelin Oluşturulması*” adlı aktiviteye kontrol inputu olarak gönderilir.



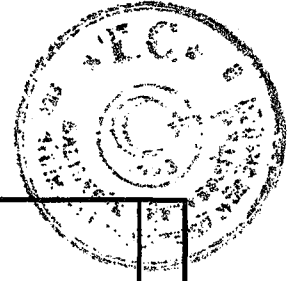
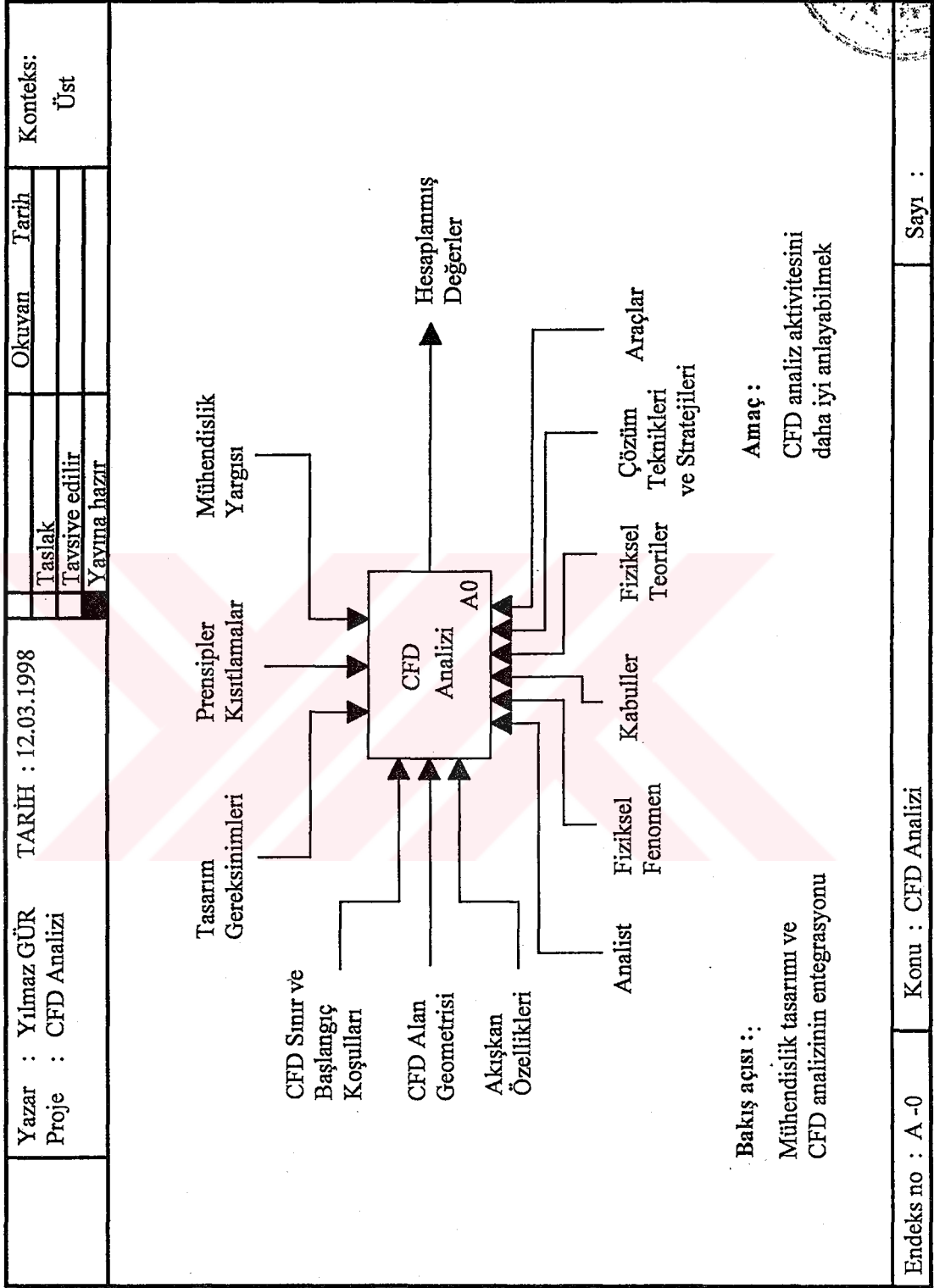
Shepard ve Wentorf [73] tarafından hazırlanan çalışmadan da görüleceği üzere idealleştirme hatalarının kontrol edilmesi konusunda değişik teknikler mevcuttur. Hali hazırda mevcut olan ve idealleştirme hatalarının kontrolunda kullanılan teknikler ise aşağıda temel aldıkları konulara göre sıralanmışlardır;

1. analitik tabanlı hata tahmini uygulaması,
2. bir hiyerarşik modelin uygun bir karşılaştırma metodolojisi ile oluşturulması,
3. eldeki mevcut duruma benzeyen bir başka idealleştirilmiş probleme elde edilen sonuçların uygulanması, örneğin analitik olarak çözülmüş bir konfigürasyona ait sonuçların kullanılması gibi.
4. ilgi duyulan performans değerleri üzerine en fazla yada en az etki yapan idealleştirme adımlarının hassasiyet analizi uygulaması ile belirlenmesi,
5. arzu edilen performans parametrelerinin değişimlerini belirleyecek istatistiksel modelleme tekniklerinin uygulaması,
6. elde edilen sonuçların bilinen fiziksel limitlere, test sonuçlarına veya makul derecedeki fiziksel limitlere uygulanmak suretiyle karşılaştırılması,
7. eksper girdi veya güncel uygulama tabanlı kuralların uygulanması.

Eğer, ilk etapta veya idealleştirme hatalarının düzeltilmesinden sonra elde edilen değerler kabul edilebilir ise o takdirde “*Hesaplanmış Özellik Değerleri*” aktiviteyi çıktı olarak terketmektedir.

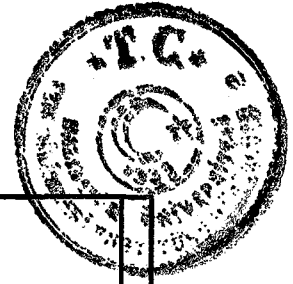
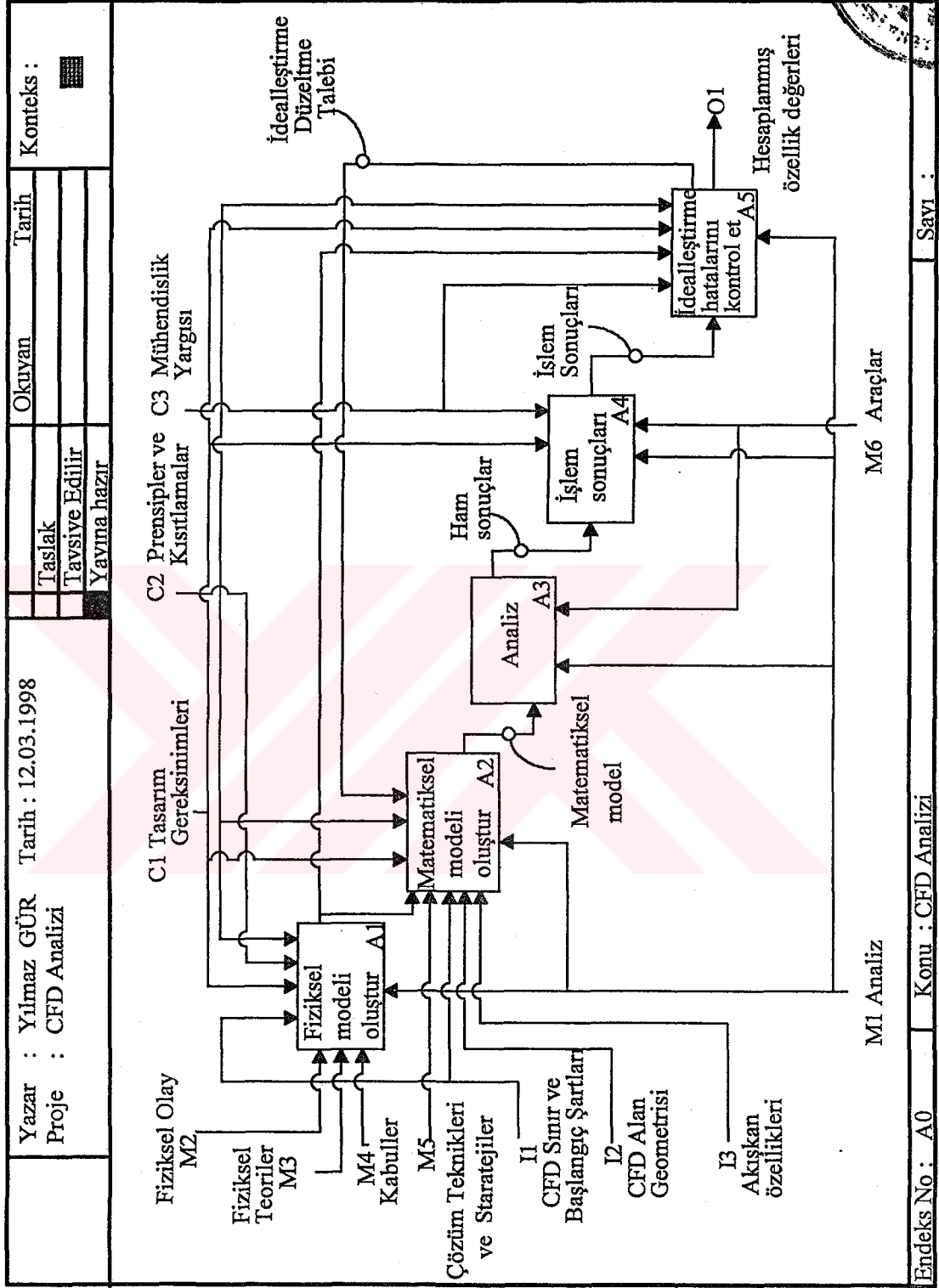
#### 4.4 Bölümsel Sonuç ve Değerlendirmeler

Bu bölümde, bilgisayar destekli akışkanlar dinamiği analizinde yer alan aktiviteler araştırılıp, bunlar SADT aktivite modelleme yöntemi ile modellendi. Öncelikle modelleme tekniği hakkında kısa bir bilgi verildi. Aktiviteler için gerekli olan girdiler, aktivitelerin yerine getirilmesine araç olan mekanizmalar, aktiviteleri kontrol eden kontrol bilgileri ve aktivite sonucunda elde edilen çıktılar, takip eden sayfalarda grafik olarak ifade edilirken bölüm içerisinde de her diyagramın metin olarak açıklandı.

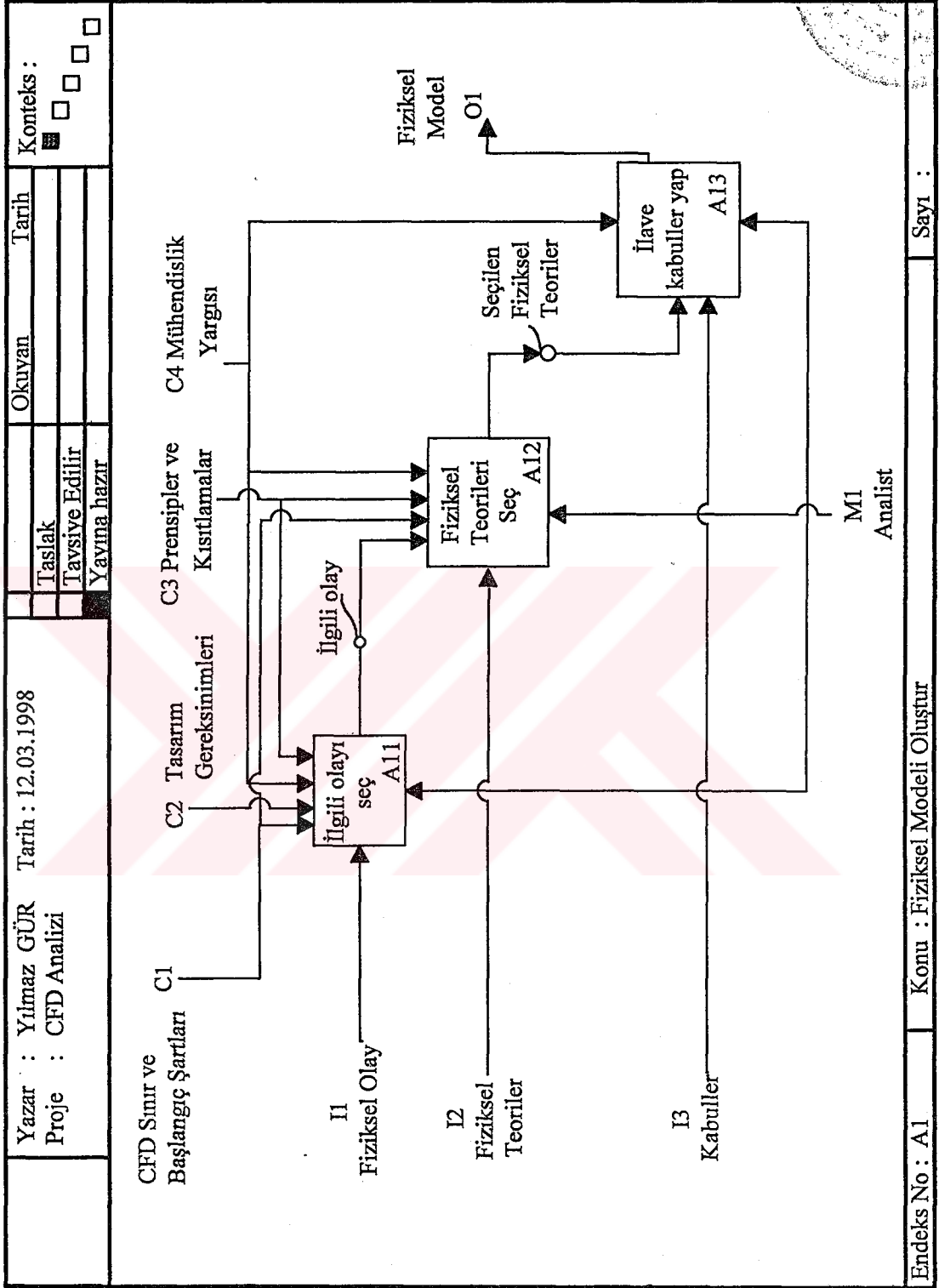


Şekil 4.2 CFD analizi (Konteks) ; Endeks No A-0

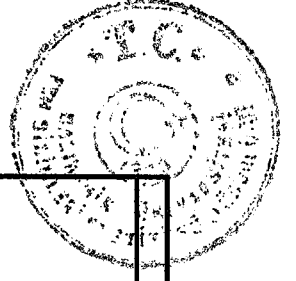
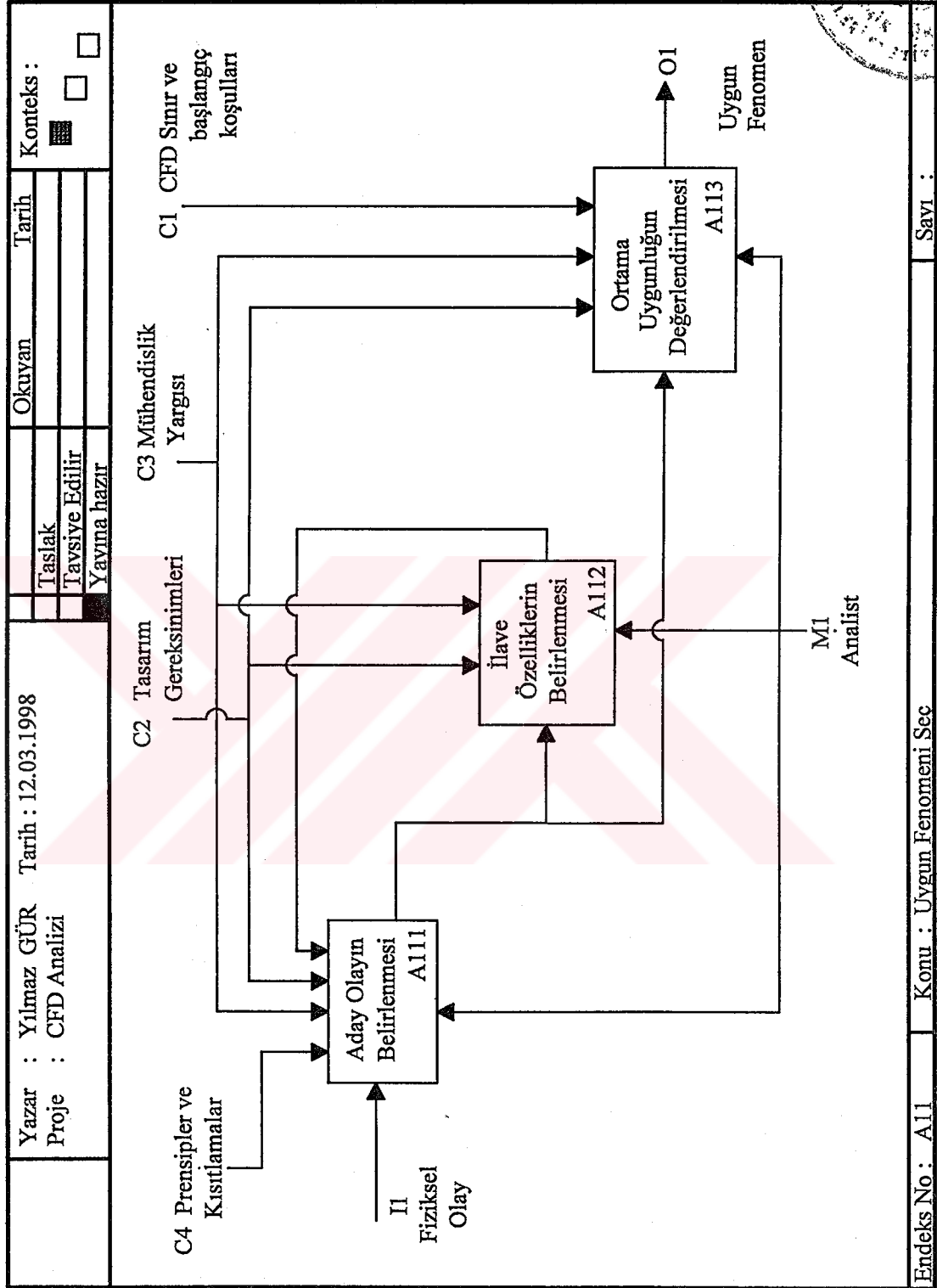




Şekil 4.3 CFD analizi; Endeks No A0

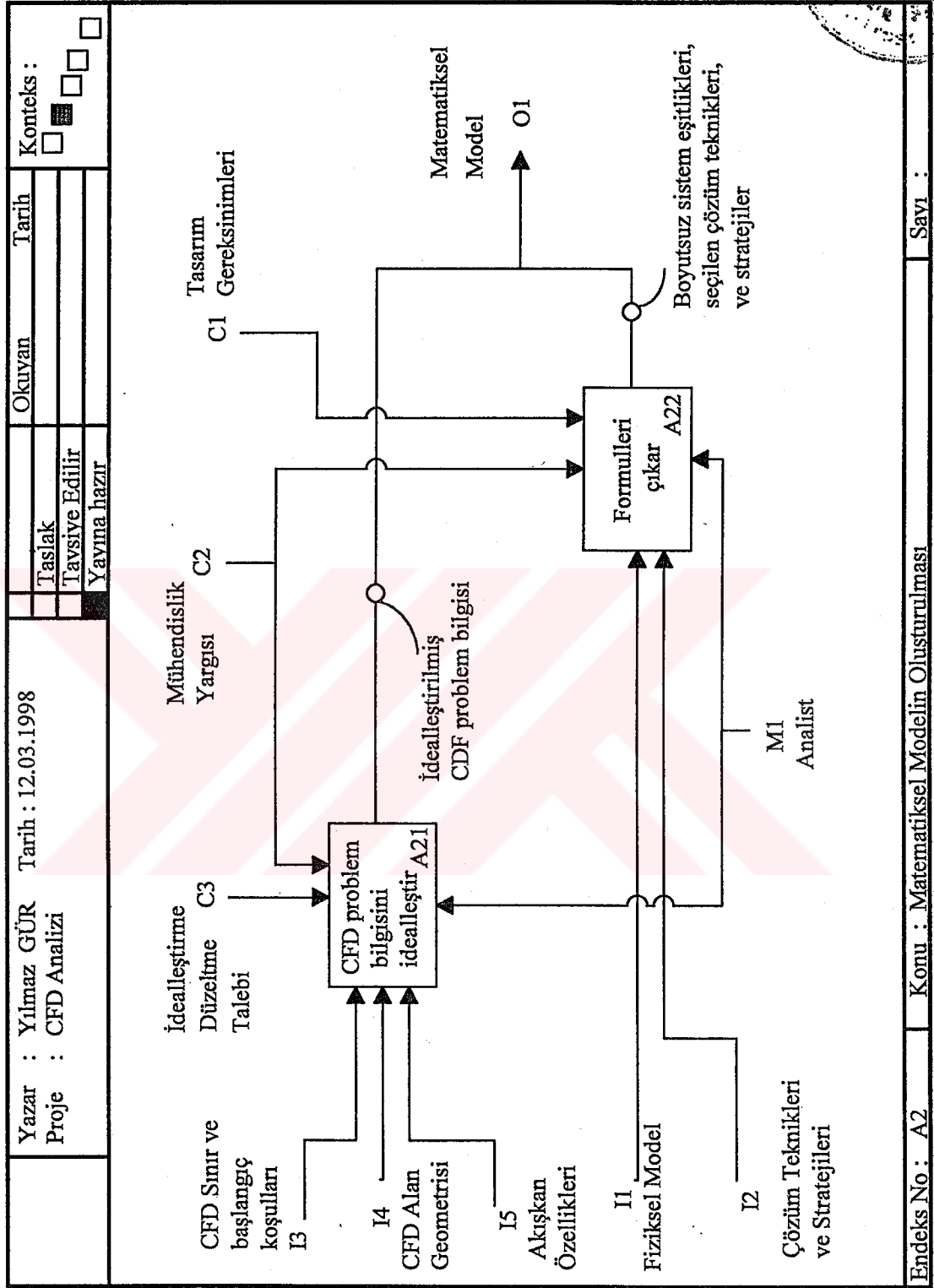
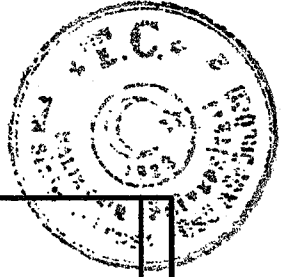


Şekil 4.4 Fiziksel modelin oluşturulması; Endeks No A1



Endeks No : A11 Konu : Uygun Fenomeni Sec Sayı :

Şekil 4.5 Uygun fenomenin seçilmesi; Endeks No A11

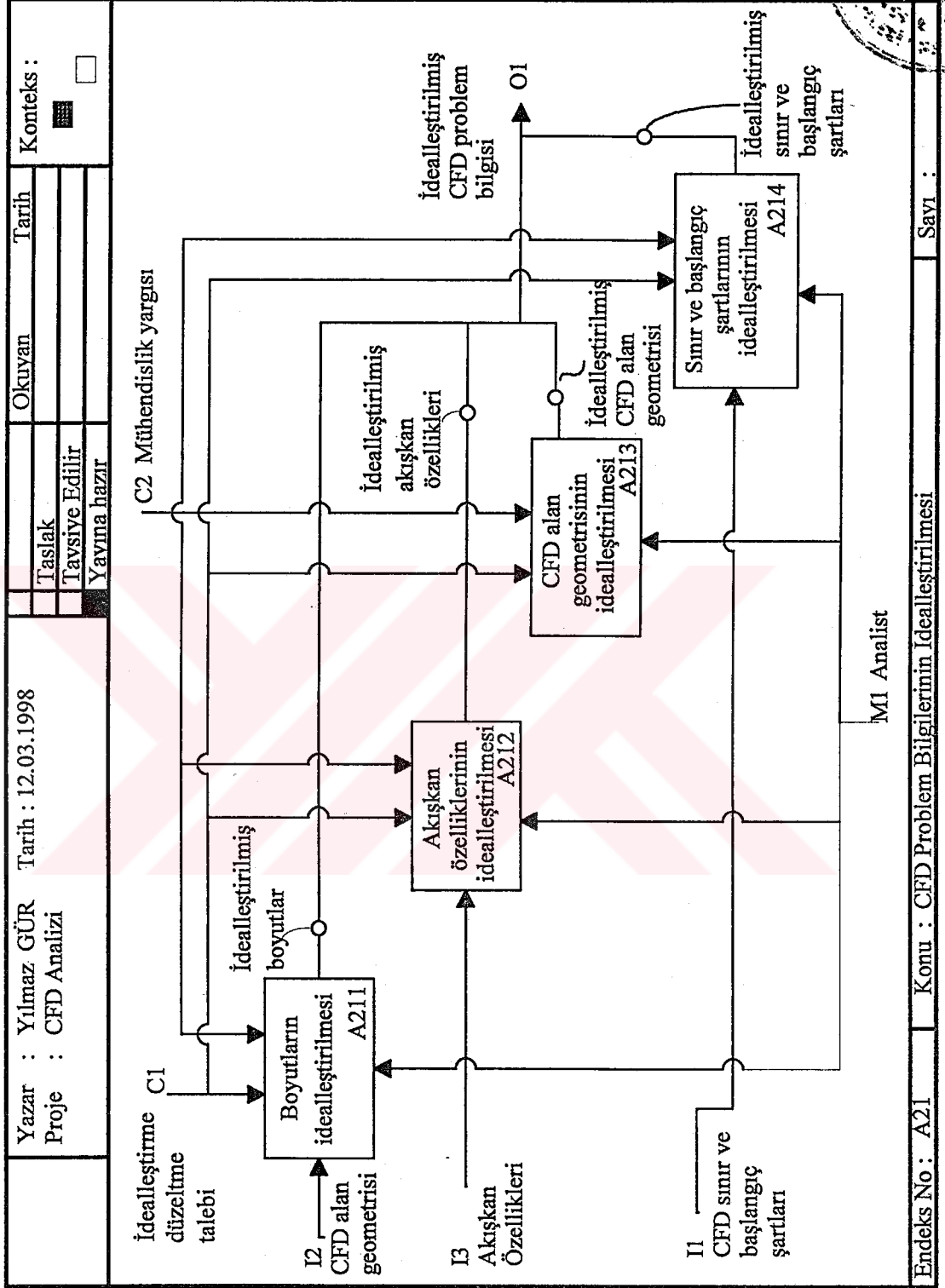


Endeks No : A2

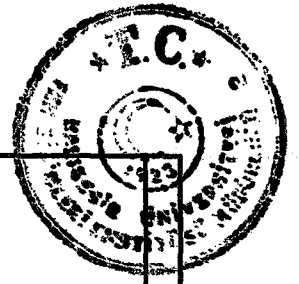
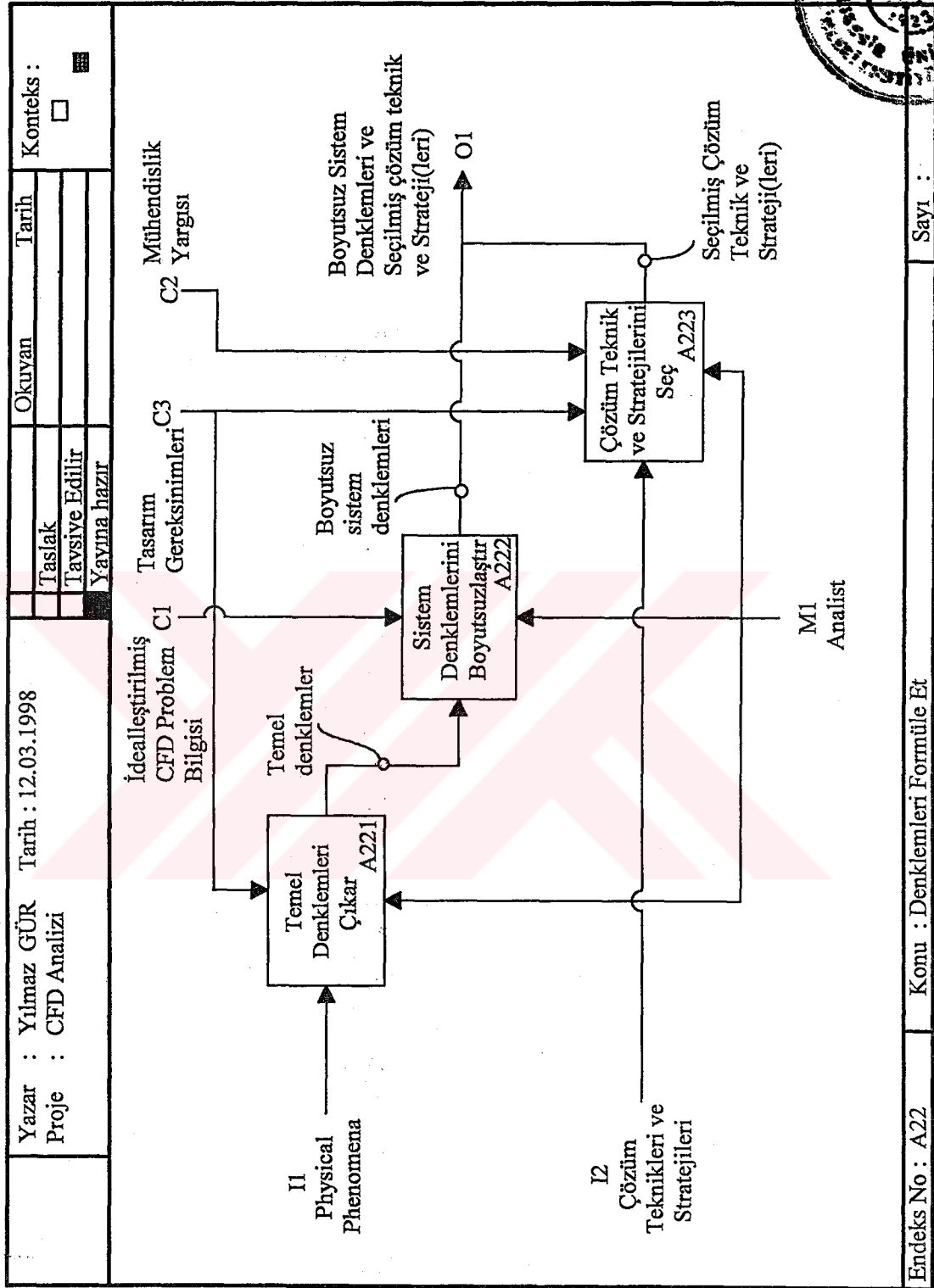
Konu : Matematiksel Modelin Oluşturulması

Sayı :

Şekil 4.6 Matematiksel modelin oluşturulması; Endeks No A2

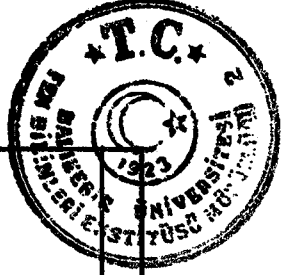
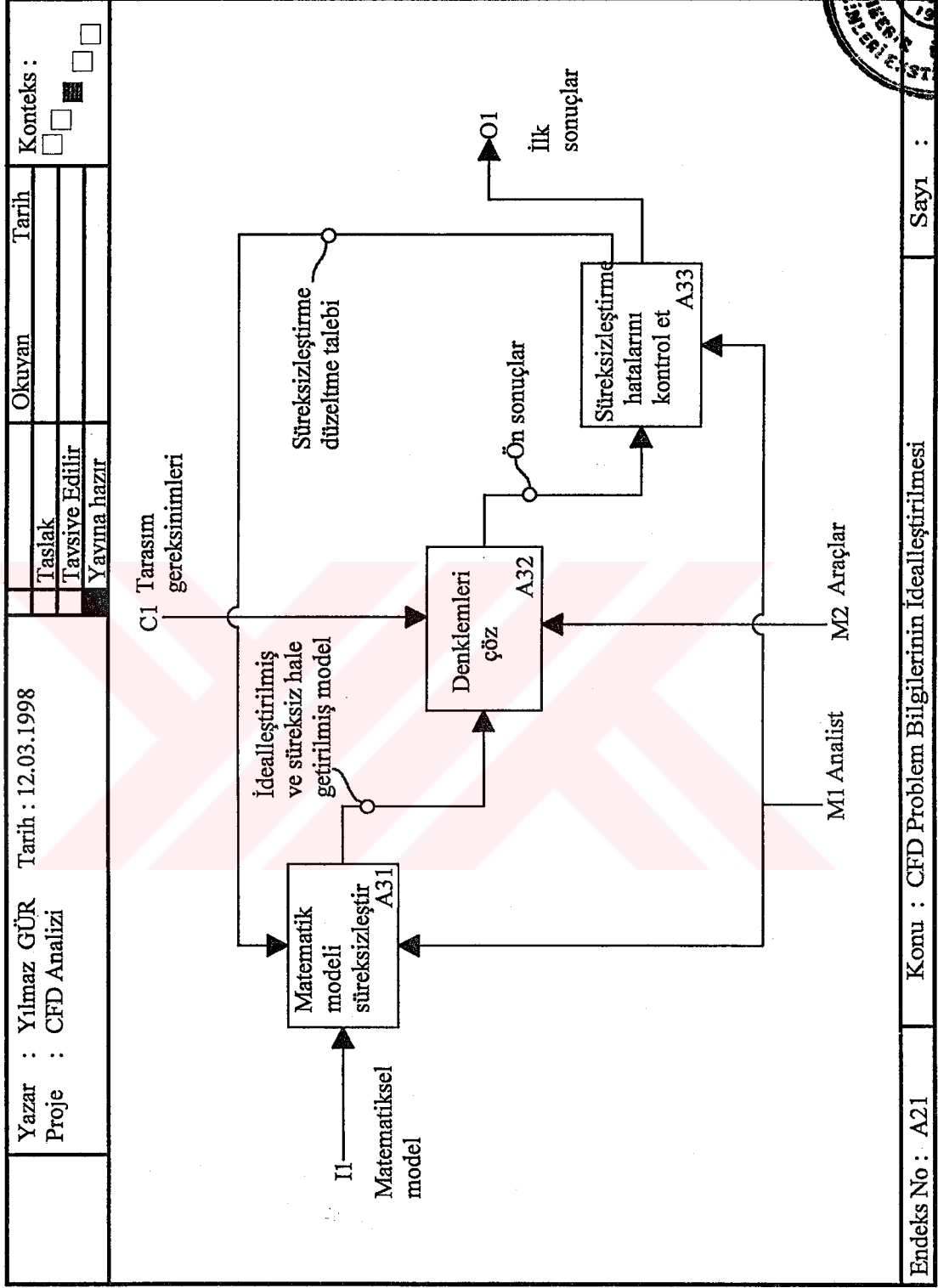


Şekil 4.7 CFD problem bilgilerinin idealleştirilmesi; Endeks No A21



Endeks No : A22      Konu : Denklemleri Formüle Et      Sayı :

Şekil 4.8 Denklemleri formüle et ; Endeks No A22



Sayı :

Konu : CFD Problem Bilgilerinin İdealleştirilmesi

Endeks No : A21

Şekil 4.9 CFD analiz aşaması; Endeks No A3



## **5. BİLGİSAYAR DESTEKLİ AKIŞKANLAR DİNAMİĞİ (CFD) ANALİZİ İÇİN BİR BİLGİ MODELİ**

### **5.1 Giriş**

Bu bölümde, CFD analizinin mühendislik dizayn prosesine integrasyonunu sağlamak amacıyla, CFD analizini destekleyecek bir bilgi modelinin, EXPRESS-G denilen grafiksel bilgi modelleme dili ile oluşturulması anlatılacaktır. Öncelikle bilgi modelleme kavramı tanıtılacak, sağlayacağı avantaj ve dezavantajlar belirtilecek ve ardından da oluşturulan bilgi modelinin tanıtılmasına geçilecektir.

### **5.2 Bilgi Modellemesi**

Bilgi, mühendislik analizi için elzem bir kavram olup, onsuz mühendislik analizi yapılamıyacağı gibi herhangi bir şekilde eksik olması dahi beklenmeyen sonuçlara neden olabilmektedir. Mühendislik analizinin esas amacı, belirli koşullar altında fiziksel sistemlerin vereceği cevaplar hakkında bilgi elde etmektir [74].

Bu bölümün amacı ise, CFD analizi için gerekli olan bilgilerin araştırılması ve bilgi modelleme tekniklerinden birini kullanarak modellenmesidir. Fakat CFD analizi için gerekli bilgileri belirlemeden önce bilginin ne olduğu, bilgi modellemenin amacı ve modelleme tekniklerinin tanımlanması üzerinde kısaca durmak faydalı olacaktır.

Bilgi, iki veya daha fazla kişi arasında haberleşmek amacıyla kullanılan bir araç olup aynı zamanda bunlar arasında da transfer edilebilir. Bilgi modellemesinin esas amacı ise, gerçek hayattaki bilgilerin, bunların kaynağı hakkında herhangi bir bilgi sahibi olmaksızın veya herhangi bir kabul yapmaya gerek kalmaksızın bilgisayar aracılığıyla



işlenmesi ve etkin bir biçimde iletişim sağlanması amacıyla bu bilgilere ait betimlemeleri formüle etmektir [75]. Bilimsel anlamda bir *model*, sistemin performansı üzerinde etkili olabilecek sistemdeki bir takım değişikliklerin etkilerini tahmin etmek üzere kullanılan, sistemin bir tasviridir[76]. *Bilgi modeli* ise, birlikte gerçek hayatın ilgi duyulan kısmına ait bir modeli oluşturan ve net bir şekilde bir dizi yorumlama kuralları sağlayan, düşünce, olgu ve proses tiplerinin formal bir biçimde tanımlanmasıdır[75].

Bir bilgi modelinin yaratılmasının amacı, bilgisayar ortamında kolayca kullanılabilen bir bilgi sisteminin tasarlanması ve oluşturulmasıdır. Bir bilgi modelinin yapılması, bize bir aktiviteyi başarılı olarak gerçekleştirebilmek için gerekli bilgileri gösterecektir.

### 5.2.1 Bilgi Modellemenin Faydaları

Bilgi modellemesinin faydaları aşağıda olduğu gibi maddeler halinde sıralanabilir [76];

- amaçlar ve bilgi gereksinimleri arasında ilişki temin etmeye yarayan dataların detaylı olarak analiz edilmesi suretiyle, gereksinimlerin detaylı olarak ortaya çıkarılması için bir çerçeve çalışma temin eder,
- işlemler, okunup anlaşılması zor ve karmaşık olan özelliklerin yerine, basit diyagramlar ve tanımlarla temsil edildiğinde anlaşılmayı kolaylaştırır,
- yapılan kabuller ve yapı gözle görülebilir ve açık, dolayısıyla da kolaylıkla üzerinde tartışılabilir ve doğruluğu kontrol edilebilir olduğundan riski azaltır,
- aktiviteleri daha başarılı bir şekilde ve kesin olarak gerçekleştirebilmeyi sağlayacak imkanların bulunması ve değerlendirilmesi için bir çerçeve çalışma sağlar,
- yapısal kararlar sadece bir kez verilir ve geliştirme süreci boyunca aynen kullanılır.



### 5.2.2 Bilginin Temsil Edilmesi

Bilginin herhangi bir belirsizlik taşımayacak bir biçimde mübadelesi için, üzerinde anlaşmaya varılmış olan yorumlama ya da açıklama kuralları ile bilgiyi temsil eden verilere ihtiyaç bulunmaktadır.

Daha önce de bahsedildiği gibi bir bilgi modeli formal bir tanımlamaya sahiptir. Bu tanımlama iki biçimde olabilmektedir;

1. yazılı bir dil kullanarak (kelimeler ile ifade)
2. şekiller kullanarak (grafiksel ifade).

Kelimeler ile ifade tarzında nesnelere, semboller ve bir takım kurallarla ifade edilirler. Diğer yandan grafiksel ifade tarzında ise, nesnelere ifade edilmesinde ikonlar ve bağlantılar kullanılmaktadır.

### 5.2.3 Nesnelere Grafiksel Olarak veya Kelimelerle İfade Etmenin Avantaj ve Dezavantajları

Grafiksel ifade biçiminde, bir modeli anlatılan kişilere açıklamak ve ayrıca modeli ifade etmek üzere dikdörtgenler, ilişki çizgileri çizmek ve bağlantıları kontrol etmek çok daha kolaydır. Fakat model git gide karmaşık hale geldikçe, modeli oluşturmak ve bağlantıları kontrol etmek zorlaşmaktadır. Grafiksel ifade tarzının tehlikesi ise, doğru görünmesine rağmen bazen yanlış olabilme ihtimalinin olmasıdır. Ve diğer bir sakıncalı tarafı da, bilgisayar tarafından tanınıp işleme konamamasıdır.

Diğer yandan, kelimelerle ifade tarzı, kurallarla resmi olarak tanımlanmış bir anlama ve kurallarla belirlenmiş bir bilgisayar diline ve aynı zamanda bir matematiksel zemine sahiptirler. Bu tür ifade tarzının diğer iyi yanı ise bilgisayar tarafından değerlendirilebilir olmalarıdır. Fakat ne yazık ki, kelimesel ifade biçimi, kolaylıkla başkalarına izah edilemezler ve izlenmeleri de bir o kadar zordur.



### 5.3 CFD Analizi İçin Bir Bilgi Modeli

Bu bölümde, CFD analizi için önerilen bir bilgi modeli sunulacaktır. Bu bilgi modelinin amacı CFD analiz paketi için gerekli olacak dataların belirlenmesidir. CFD analizi için gerekli olan bilgilerin belirlenmesine yönelik bu model, 5.1 ... 5.9 nolu şekillerle ifade edilmiş ve gerekli bilgileri ile ilgili açıklamalar ise aşağıda verilmiştir.

Bilgi modeli, grafiksel bir bilgi modelleme dili olan ve EXPRESS-G olarak adlandırılan bir dil kullanılarak oluşturulmuştur. EXPRESS-G ile ilgili daha ayrıntılı bilgi Ek-B'de bulunabilir.

#### 5.3.1 Fiziksel Model

Bir *Fiziksel Model*, fiziksel teorileri ve o probleme özgü kabulleri içeren fiziksel bir problemin boyutlardan bağımsız bir biçimidir. Bir fiziksel model, belirli bir problem için birden fazla fiziksel teoriyi de içerebilir. CFD analizi için bilgi modelinde yer alan fiziksel modele ait kısım Şekil 5.1'de gösterilmiştir.

##### 5.3.1.1 Fiziksel Fenomen

Fiziksel fenomen, fiziksel bir problemin doğal davranışının sınıflandırılmasını tanımlamaktadır. Örneğin problemin bir CFD problemi veya ısı transferi problemi veya benzeri olarak sınıflandırılması gibi.

##### 5.3.1.2 Fiziksel Teoriler

Fiziksel teoriler, fiziksel bir fenomeni modellemek için kullanılacak genel bir yaklaşımı ifade etmektedir. Bunlar, gerek formüle edilmiş biçimde veya gerekse

sembolik olarak terimlerle ifade edilebilen temel fiziksel kanunlardır. Örneğin, Newton'un kütleinin korunumu kanunu, enerjinin korunumu kanunu ve benzeri gibi.



### 5.3.1.3 Kabuller

Kabuller, fiziksel fenomen ile model arasındaki farklar olarak da ifade edilebilir. Genel olarak bunlar, fiziksel bir fenomene onu daha basit, belirli bir doğruluk derecesine kadar çabuk çözülebilir, ve kolaylıkla yönetilebilir hale getirmek üzere tanıtılan basitleştirmelerdir. Genellikle, basitleştirmeler ne kadar büyük ve yapılan kabuller ne kadar çok ise, oluşturulan model de o kadar gerçekten uzaklaşmaktadır. Eğer bir fiziksel fenomeni olduğu gibi analiz etmek mümkün olsaydı, o takdirde herhangi bir kabul yapmaya da gerek olmayacaktır.

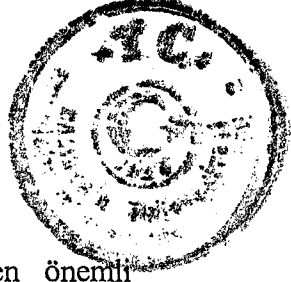
Bazı durumlarda, ilk adımda mümkün olduğu kadar çok kabul yapmak ve daha sonra bunları birer birer kaldırarak analiz yapmak da faydalı olabilmektedir. Bu yolla, mühendis, ilk etapta kaba fakat çabuk bir çözüme ulaşır ve daha sonra istenilen hassasiyete ulaşıncaya kadar yapılan kabulleri azaltarak problemi iyileştirebilir[15].

### 5.3.1.4 Mühendislik Yargısı

Mühendislik yargısı, ya analizi yapan kişinin deneyiminden veyahutta önceki mühendislik deneyimlerinden gelen bilgileri ifade eder. Tasarımcı yapmış olduğu her analizden yeni bir deneyim kazanır. Bu deneyim bilgilerinin üst üste biriktirilmesi tasarımcının tecrübesini teşkil eder.

### 5.3.1.5 Prensipler ve Sınırlamalar

Prensipler ve sınırlamalar bir çözümün teorik ve pratik yönünü tanımlamaktadır [77]. CFD analizi bilgi modelinin Prensipler ve Sınırlamalar ile ilgili kısmı Şekil5.9'da



gösterilmiştir.

Bir tasarım problemi belirli sınırlar içerisinde mevcuttur. Örneğin en önemli sınırlamalardan bir tanesi ürün maliyetidir. Bu sınırlama, üretimi isteyen kişi ya da firmanın ne kadarlık bir harcama yapmaya hazırlıklı olduğuna veya o ürünü alacak müşterilerin ne ödemeyi beklediğine bağlıdır. Diğer sınırlamalar ise, örneğin, o ürünün ağırlığı veya boyutları da olabilir. Bir başka türdeki sınırlama ise, standartlar tarafından belirlenen emniyet kuralları veya yasalarla belirlenen bazı kurallar olabilir.

Yapılan tasarım tarafından neyin gerçekleştirilebileceğine dair konulan bir takım sınırlamalar, aynı zamanda kabul edilebilir çözümlerin alanını da sınırlandıracaktır. Bu aynı şekilde tasarımcının hedeflediği alanı da daraltacak ve hatta bir çok kabul edilebilir çözüm dahi gereksiz olarak elenmek zorunda kalacaktır.

Öte yandan, sınırlamaların daha geniş veya üstü kapalı tutulması ise tasarımcıyı uygun olmayan bir yöne doğru sürükleyecek ve dolayısıyla da uygun olmayan bir takım çözümlere ulaşılacaktır.

Premsiplerin ve sınırlamaların tasarım prosesinin ilk etaplarında tanımlanması, bazı sınırlamalar getirecek ve bunlar önerilen çözümlerin kabul edilebilir sınırlar içerisinde kalıp kalmadığının yeniden gözden geçirilip değerlendirilmesinde de kullanılabilir [78].

### 5.3.2 Matematiksel Model

CFD analizi için oluşturulan matematiksel model, matematiksel denklem takımlarını, analizi yapılacak CFD alanının geometrik tanımını, problemin sınır ve başlangıç koşullarını, akışkan özelliklerini, matematik denklemleri nümerik olarak çözmek için çözüm tekniklerini, analizin amacını gerçekleştirmek üzere çözüm stratejilerini içerir. Matematiksel model, aynı zamanda matematiksel modelin oluşturulmasında kontrol görevini üstlenen Mühendislik Yargısı ve Tasarım Gereksinimleri ile de yakından

alakalıdır. Matematiksel modelin CFD analizi için oluşturulan bilgi modelindeki yeri de Şekil 5.1’de görülmektedir.



### 5.3.2.1 Akışkan Özellikleri

Akışkan özellikleri belirli bir sıvının veya gazın davranışlarını belirleyen fiziksel niceliklerdir. Bu özellikler Şekil 5.4’de de gösterildiği ve aşağıda olduğu gibi sınıflandırılabilirler.

- Fiziksel özellikler (örneğin yoğunluk, vizkozite)
- Termal özellikler (örneğin ısı iletimi, özgül ısı,)
- Kimyasal özellikler (örneğin geçirgenlik, konsantrasyon)
- Mekaniksel özellikler (örneğin sıkıştırılabilirlik, elastisite modülü)
- Elektriksel özellikler (örneğin iletkenlik)

Yoğunluk ve vizkozite CFD analizi için çok önemli olan fiziksel özelliklerdir. Bir akışkanın yoğunluğu problem boyunca sabit olmayabilir, çünkü yoğunluk aynı zamanda sıcaklığa ve basınç farklılıklarına da bağlıdır. Eğer sıcaklık artarsa, moleküller aktivite ve moleküller arası mesafe de artar ve dolayısıyla da yoğunluk azalır. Aynı şekilde, eğer basınç artarsa bu kez de moleküller belirli bir hacme sıkıştırılmaya zorlanır ve yoğunluk da buna paralel olarak artar.

Tüm gerçek akışkanlar vizkoziteye sahiptir ve bir hareket söz konusu olduğunda belirli bir sürtünme fenomeni gösterirler. Vizkozite ise esasen akışkan tabakalar arasındaki kohezyondan meydana gelir ve basıncın artmasıyla birlikte az da olsa bir artış gösterir. Fakat bu değişiklik bir çok mühendislik probleminde ihmal edilmektedir. Bu arada, vizkozite, sıcaklığın değişimi ile büyük oranda değişmektedir. Sıvılarda, moleküller arasındaki kohezif kuvvetler ve vizkozite, sıcaklığın artmasıyla süratle azalmaktadır[79].



### 5.3.2.2 CFD Sınır ve Başlangıç Koşulları

Her ürün belirli bir çevre koşulu altında analiz edilir. Bu çevre koşulları, ürünün analizi işlemi üzerine, sınır ve başlangıç koşulları aracılığıyla etki etmektedir.

Eğer, problemi daha basit hale getirmek ve daha çabuk sonuç elde etmek amacıyla sınır ve başlangıç koşulları üzerine herhangi bir kabul yapılmışsa, bu durumda bunlar “*idealleştirilmiş sınır ve başlangıç koşulları*” olarak adlandırılırlar. Örneğin, hız bileşenlerinden bir tanesi (mesela kartezyen koordinat sisteminde y yönündeki hız bileşeni) sıfır olarak kabul edilirse, bu en azından bir sınır şartının idealleştirildiği anlamına gelmektedir.

Analizin yapılacağı alanın sınırlarının her bir dilimine, uygun olan sınır şartlarının uygulanması gerekmektedir[80]. Sınır koşulları uygulanmaması durumunda, sistem denklemleri tam olarak belirlenmiş olamaz. Bu aynen diferansiyel denklemlerin çözümünde yer alan sabitleri yeterli sınır şartı olmadan bulmaya benzemektedir [24].

Başlangıç koşulları sıcaklık, hız veya zaman vb. için belirlenebilir. Bu koşullar, kararlı hal simülasyonunda iteratif çözüm işlemi için bir başlangıç tahmini olarak kullanılmakta ve transiyent simülasyonda ise başlangıç anındaki sistemin durumunu belirlemektedirler [80].

### 5.3.2.3 CFD Alan Geometrisi

Üzerine matematiksel modelin uygulandığı CFD alanının geometrisi, bir mühendislik analizi aktivitesi için son derece önemlidir. Geometri ile ilgili bilgi ne kadar tam ise o alanın ifadesi de o kadar tamdır, yani herhangi bir şüpheye yer vermeyecek mahiyettedir. CFD alan geometrisi ile ilgili bilgi modeli kısmı Şekil 5.2’de verilmiştir.



CFD alanının idealleştirilmesi, yapılacak olan belirli bir analiz için gerekli olmayacak bazı tasarım özelliklerini ortadan kaldırır ve mekana ve/veya zamana bağlı sınır koşullarını sisteme ilave eder.

Matematiksel model, sistem denklemlerinden boyutların kaldırılması ve parametrelerle değiştirilmesi suretiyle de basitleştirilebilir. Örneğin, herhangi bir geometrik boyutun kaldırılmasıyla üç boyutlu bir analiz problemi iki boyutlu hale indirgenebilir.

#### **5.3.2.4 CFD Alan Geometrisinin Tanımlanması**

Geometrik bilginin düzeyi, geometriyi ifade etmekte kullanılan yöntemle bağlıdır. Geometrinin ifade edilmesinde kullanılan üç değişik yöntem mevcuttur. Bunlar tel kafes, yüzey ve katı modelleme yöntemleridir[4]. Geometrinin tanımlanması ile ilgili bilgi modeli Şekil 5.3'de verilmiştir.

Bir tel kafes model, bir nesneye ait ayrıtların ve tepelerin bilgilerini içerir fakat yüzeyle ilgili herhangi bir bilgi vermez [81]. Bir yüzey modeli ise, kenarları ile sınırlanmış bir yüzey parçasının geometrik bilgilerini vermektedir. Bir katı model ise, katıların bilgisayar olarak tam ifadesini verirken aynı zamanda katı cisime ait herhangi bir geometrik özelliğin hesaplanmasına da izin vermektedir[22].

#### **5.3.2.5 Matematiksel Denklemler**

Matematiksel denklemler, spesifik durumları modellemek üzere kullanılan veya daha genel olan fiziksel teorilerin bir alt kümesi olan formüllerdir[82]. Matematiksel denklemler, fiziksel teorileri dikkate almak suretiyle fiziksel özellikler arasındaki ilişkileri ifade ederler. Bu denklemler, bilhassa bir koordinat sistemine ve boyutlara bağlıdır. Matematiksel denklemler için oluşturulan bilgi modeli ise, Şekil 5.5'de gösterilmiştir.





Matematiksel denklemler çeşitli formdaki denklemleri, örneğin kütlelerin, enerjinin, momentumun korunumu denklemleri ve benzeri gibi, içermektedir.

### 5.3.2.6 Çözüm Teknikleri

Çözüm teknikleri, sürekli formdaki bir alanı sonlu sayıdaki parametrelerle tanımlayan matematiksel denklemlerin çözümü için kullanılan nümerik metodları temsil etmektedirler. Sürekli bir alanı süreksiz hale getirmede kullanılan en yaygın teknikler ise Sınır Elemanları metodu, Sonlu Farklar metodu, Sonlu Hacimler metodu ve Sonlu Elemanlar metodudur.

### 5.3.2.7 Çözüm Stratejileri

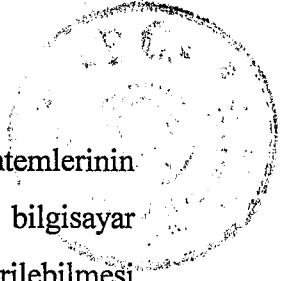
Sonlu Elemanlar metodu gibi nümerik analiz metodları, çok çeşitli tipteki problemlerin çözümü için geniş yelpazeli bilgisayar destekli çözüm stratejileri sunmaktadır. Fakat burada kullanıcı, verilen bir problem için mutlaka belirli bir çözüm stratejisi seçmek zorundadır. Eğer problem zamana bağlı bir problem ise, bu durumda her iterasyonda arttırılacak zaman aralığı da verilmelidir.

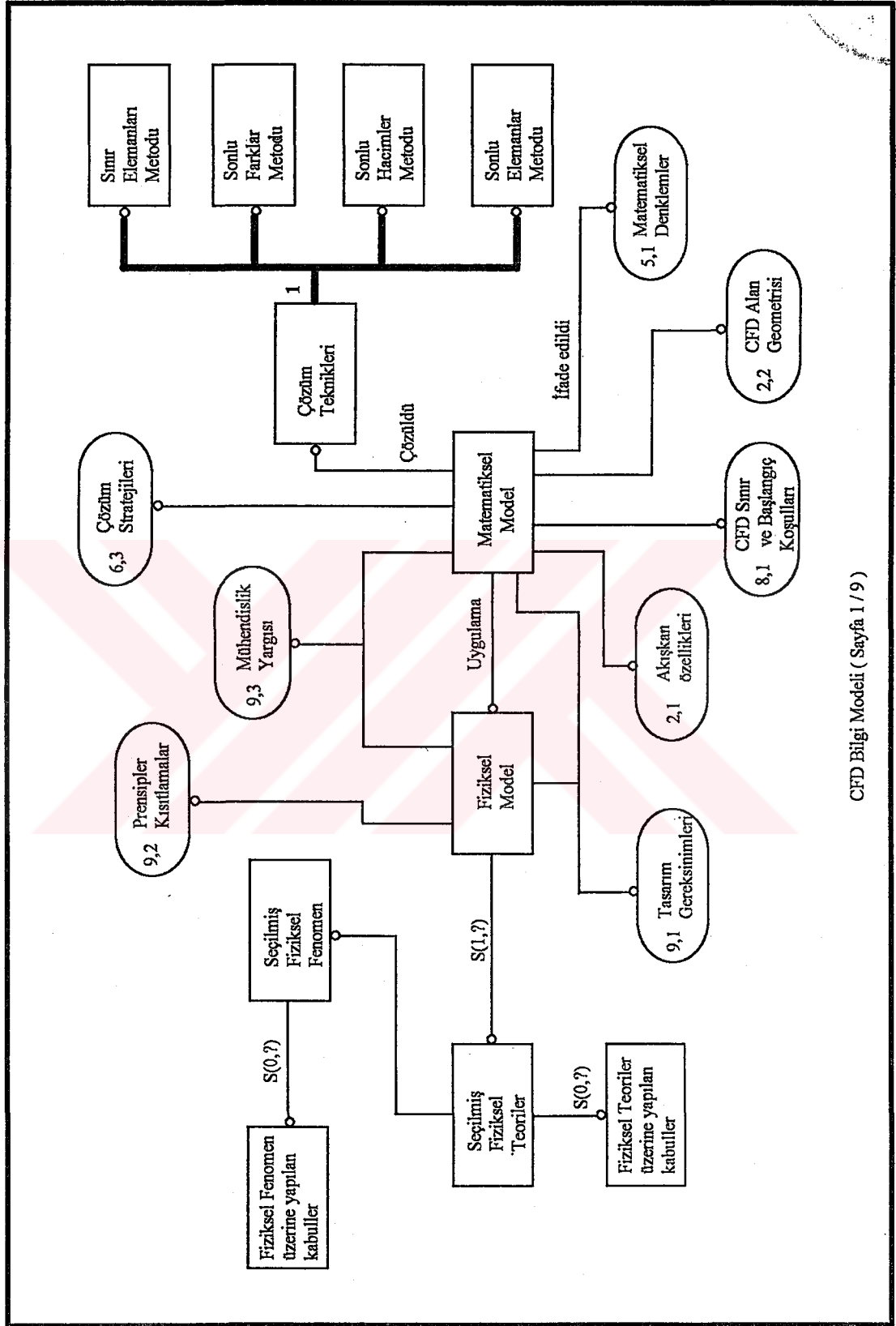
Çözüm stratejisi veya stratejilerinin seçimi ise, genellikle kullanıcının veya analizi yapan kişinin önceki deneyimlerine dayanmaktadır. Bir çok durumda, bir problem için seçilebilecek çözüm stratejisi tek değildir veya kesin bir strateji mevcut değildir. Farklı sınıflardaki problemler için, uygun bir çözüm algoritmasının seçilmesi, daha az bilgisayar zamanı ve çabası ve daha iyi sonuçlar verebilmektedir. Çözüm stratejileri ile ilgili bilgi modeli ise Şekil 5.6 ve 5.7'deki diyagramlarda gösterilmiştir.

## 5.4 Bölümsel Sonuç ve Değerlendirmeler

Bu bölümde, ilk olarak bilgi modelleme kavramı tanıtıldı, bilgi modellemesinin

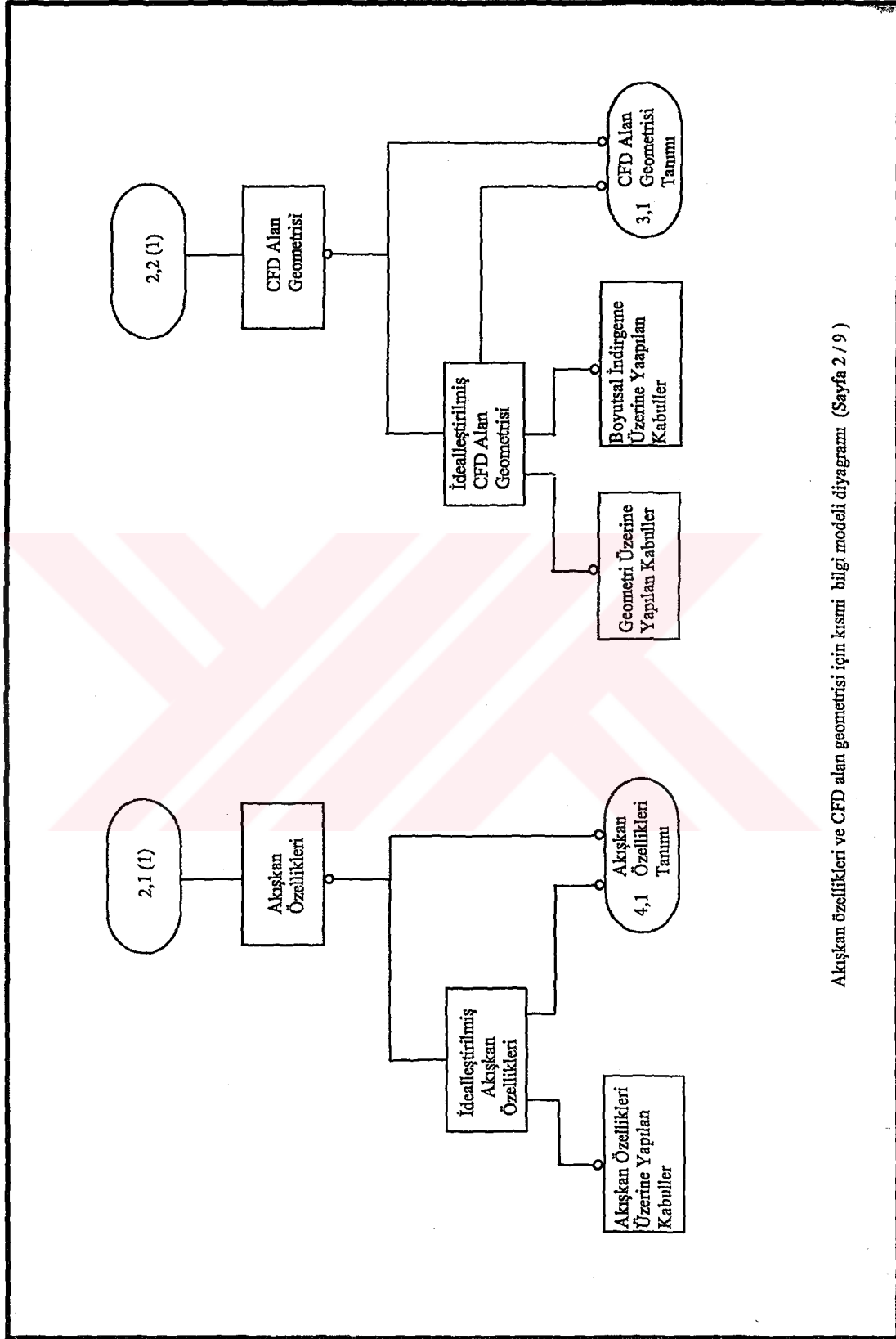
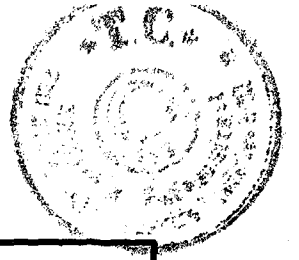
sağlayacağı faydalar incelenip, bilginin temsil edilme yöntemleri ve bu yöntemlerinin birbirlerine göre olan avantaj ve dezavantajları tartışıldı ve devamında da bilgisayar destekli akışkanlar dinamiği analizi işlemindeki aktivitelerinin gerçekleştirilebilmesi için gerekli olan bilgiler ve bunlar arasındaki ilişkiler, grafiksel bir bilgi modelleme dili olan EXPRESS-G ile modellendi ve önce modelde yer alan diyagramların açıklama metinleri verildi, bölümün son kısmında da çizilen diyagramlar toplu olarak sunuldu. .





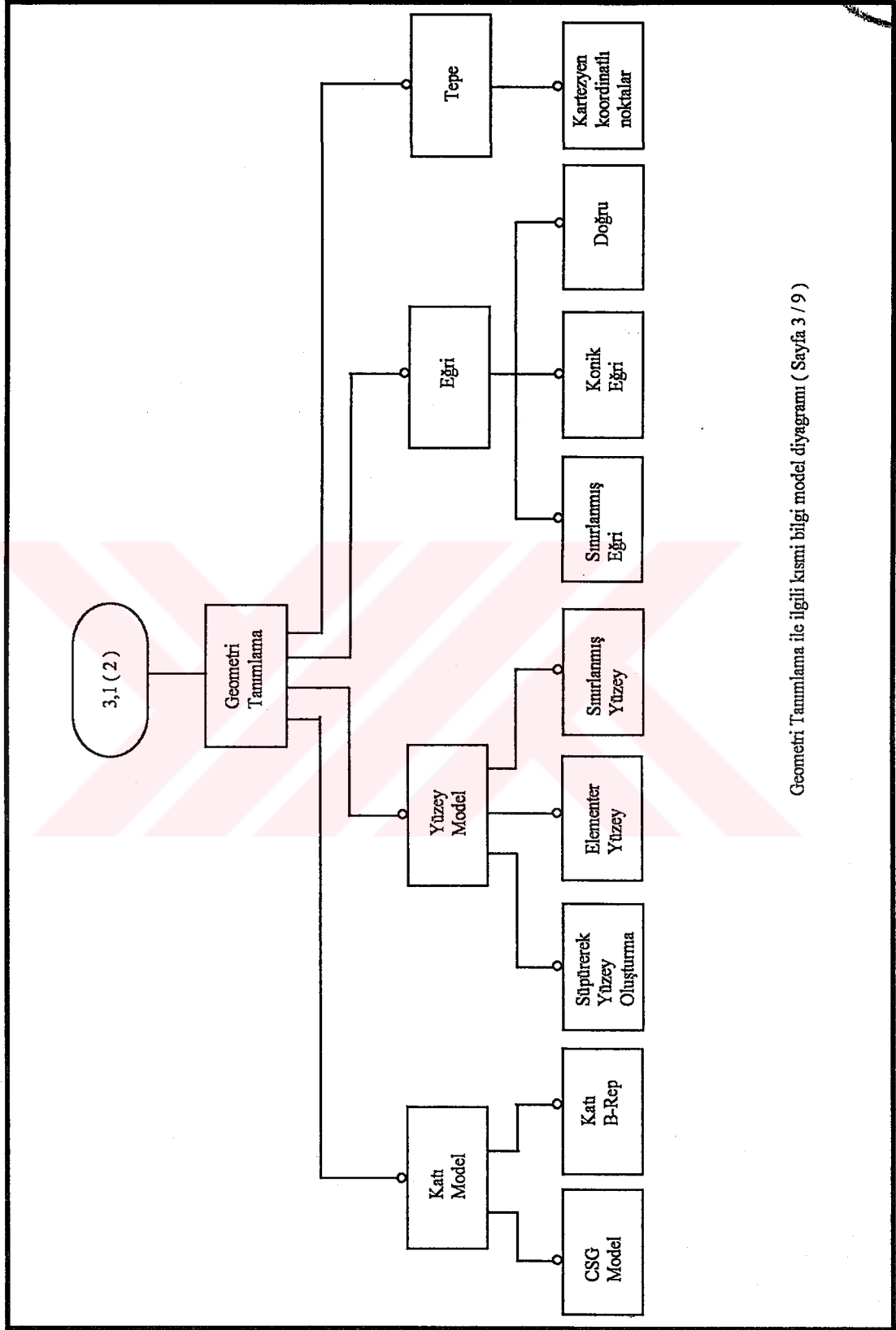
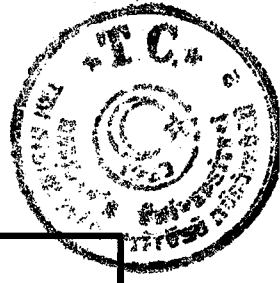
CFD Bilgi Modeli ( Sayfa 1 / 9 )

Şekil 5.1 CFD analizi için kısmi bir bilgi modeli diyagramı



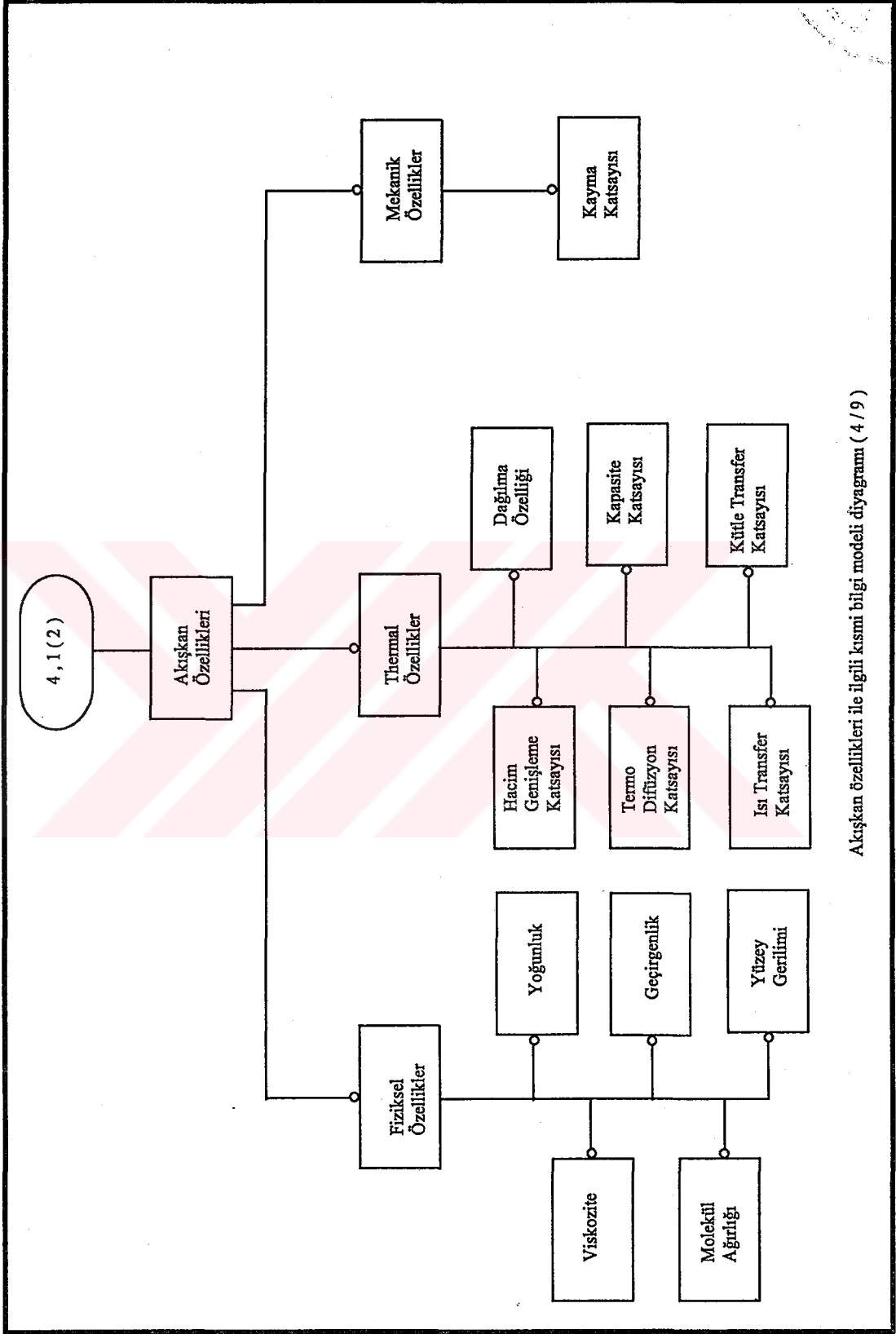
Akışkan özellikleri ve CFD alan geometrisi için kısmi bilgi modeli diyagramı (Sayfa 2 / 9)

Şekil 5.2 Akışkan özellikleri ve CFD alan geometrisi için kısmi bilgi modeli



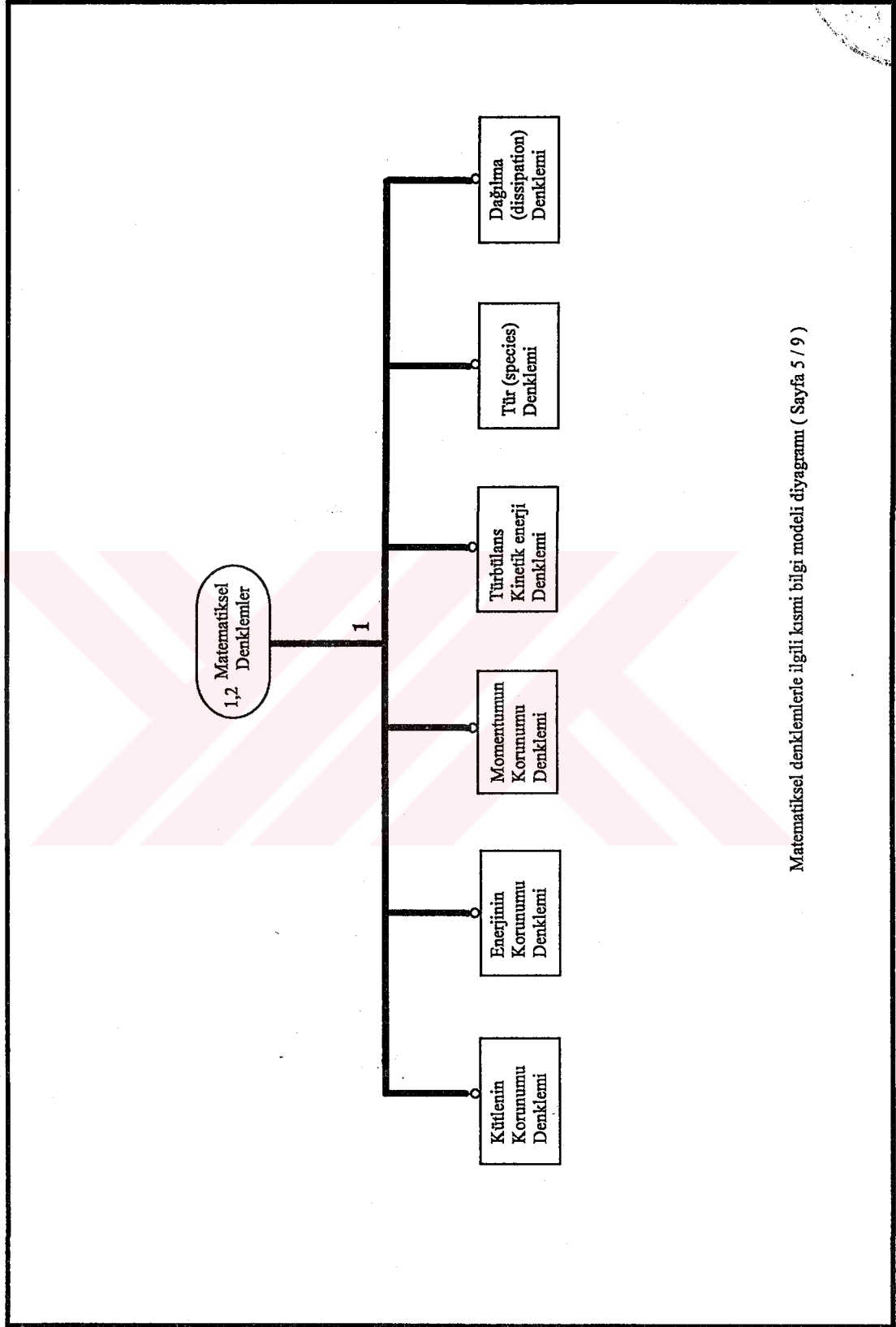
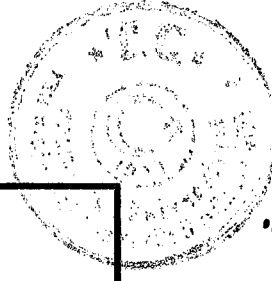
Geometri Tanımlama ile ilgili kısmi bilgi model diyagramı ( Sayfa 3 / 9 )

Şekil 5.3 Geometri tanımlama ile ilgili kısmi bilgi modeli



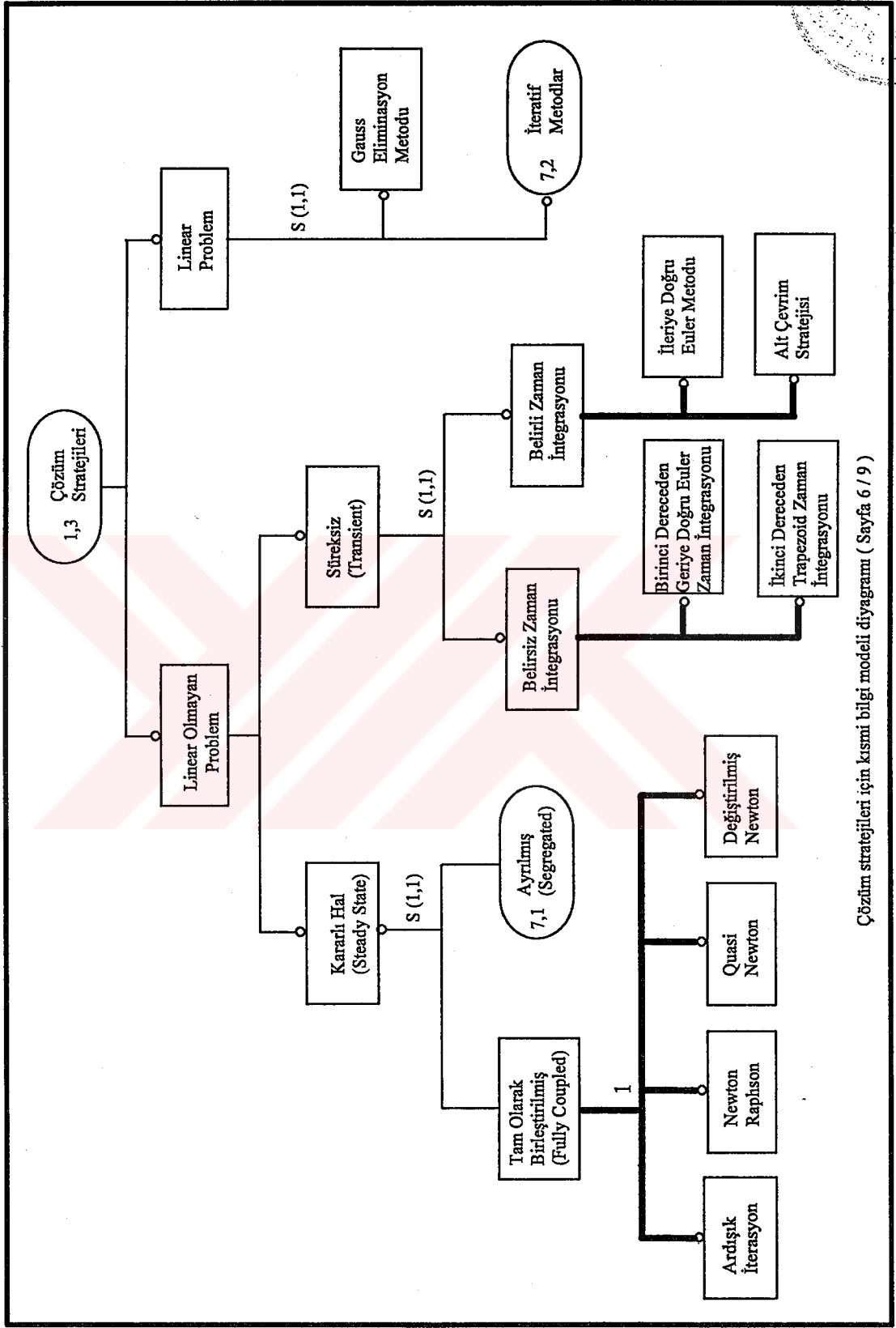
Akışkan özellikleri ile ilgili kısmi bilgi modeli diyagramı ( 4 / 9 )

Şekil 5.4 Akışkan özellikleri ile ilgili kısmi bilgi modeli



Matematiksel denklemlerle ilgili kısmi bilgi modeli ( Sayfa 5 / 9 )

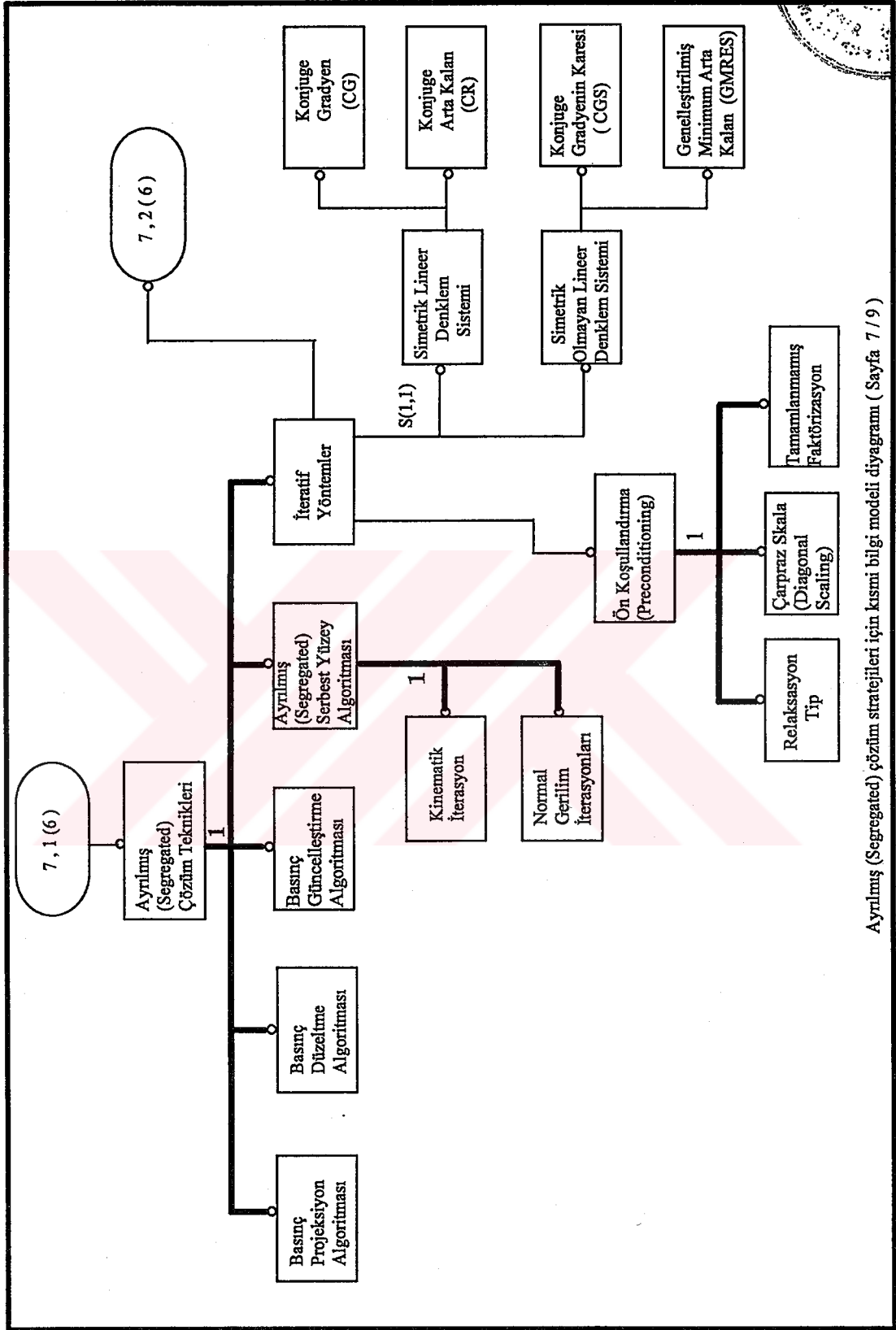
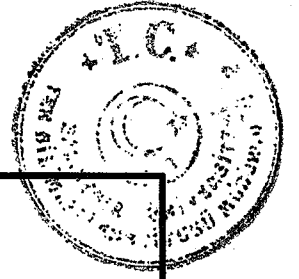
Şekil 5.5 Matematiksel denklemlerle ilgili kısmi bilgi modeli



Çözüm stratejileri için kısmi bilgi modeli diyagramı ( Sayfa 6 / 9 )

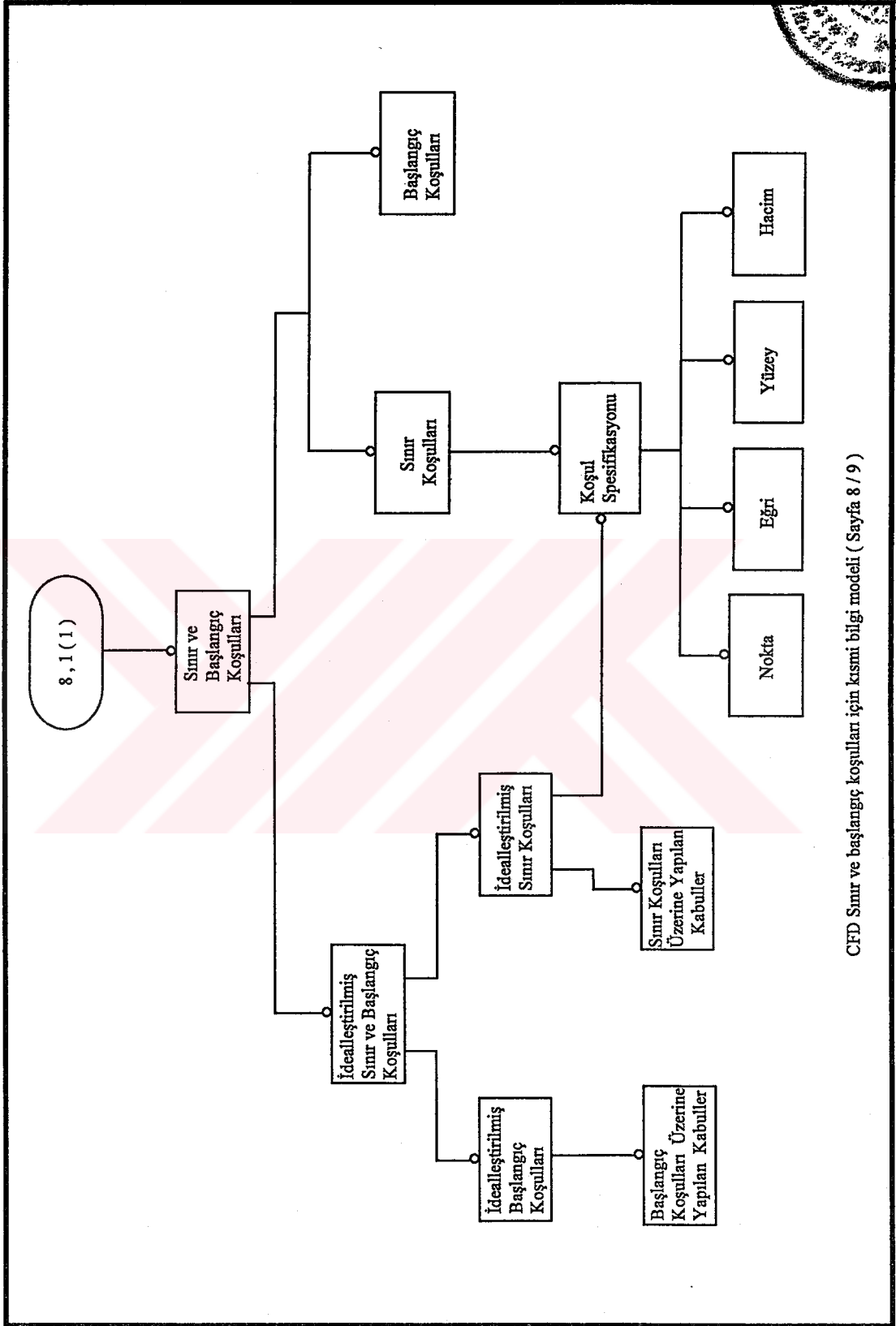
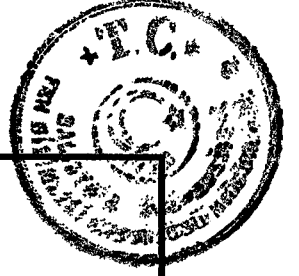
Şekil 5.6 Çözüm stratejileri için kısmi bilgi modeli





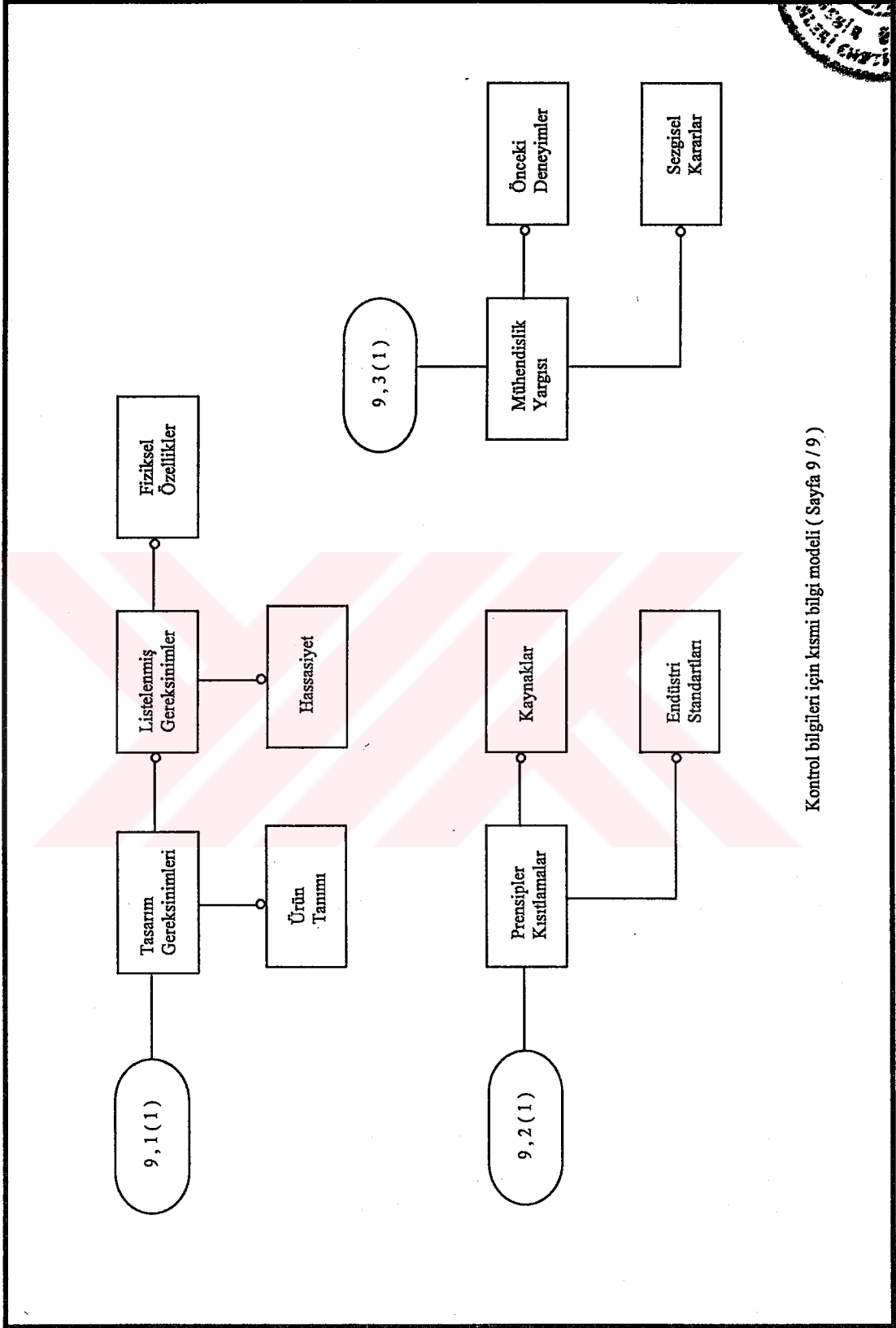
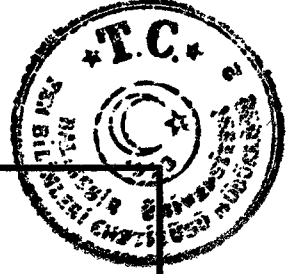
Ayrılmış (Segregated) çözüm stratejileri için kısmi bilgi diyagramı ( Sayfa 7 / 9 )

Şekil 5.7 Ayrılmış (Segregated) çözüm stratejileri için kısmi bilgi modeli



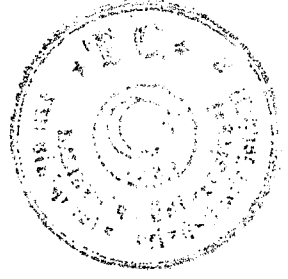
CFD Sınır ve başlangıç koşulları için kısmi bilgi modeli ( Sayfa 8 / 9 )

Şekil 5.8 Sınır ve başlangıç koşulları için kısmi bilgi modeli



Kontrol bilgileri için kısmi bilgi modeli ( Sayfa 9 / 9 )

Şekil 5.9 Kontrol bilgileri için kısmi bilgi modeli



## 6. CFD ANALİZİ DATA MODELİNİN ÜRÜN MODELLEME ORTAMINDA OLUŞTURULMASI

### 6.1 Giriş

Bu bölümde, üçüncü, dördüncü ve beşinci bölümlerde oluşturulan temeller baz alınarak, CFD analizi için gerekli olan data ve bilgilerin içinde toplanacağı ve ürün modelleme kavramı dikkate alınarak hazırlanan bilgisayar yazılımı tanıtılacaktır. Bölüm 6.2 de kullanılan bilgisayar ortamı tanıtılırken, 6.3 nolu bölümde ise CFD analizini desteklemek amacıyla oluşturulan data yapısı anlatılacak, bölüm 6.4'de oluşturulan bu data yapısı ile ürün data modeli arasında oluşturulan arayüzden bahsedilecek, ve bölüm 6.5'de de yapılan bir örnek uygulama ile yazılan program test edilecektir.

Bu bilgisayar yazılımı, aşağıda sıralanan hedefleri gerçekleştirmek amacıyla yapılmıştır:

1. Önerilen CFD analizi data modelinin yapılacak bir akışkanlar mekaniği analizinde kullanılacak dataları sağlayabileceğini ve,
2. CFD analizinin mühendislik tasarım aşamasına entegrasyonunun başarılılabileceğini göstermektir.

Adı geçen programın oluşturulması esnasında kullanılan ve **Structure Editor** denilen programın sahip olduğu bir özellikten dolayı yeni yaratılmış olan programın yapılan bir örnek uygulama ile örneklenmesinin ardından gerekli tüm dataların CFD data modeli ortamından çekilerek istenilen formatta yazdırılması sağlanmış ve bu çıktılar Ek-D'de verilmiştir.

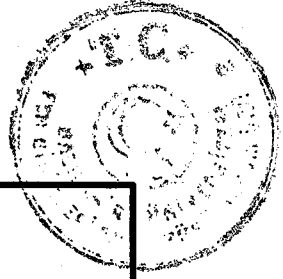
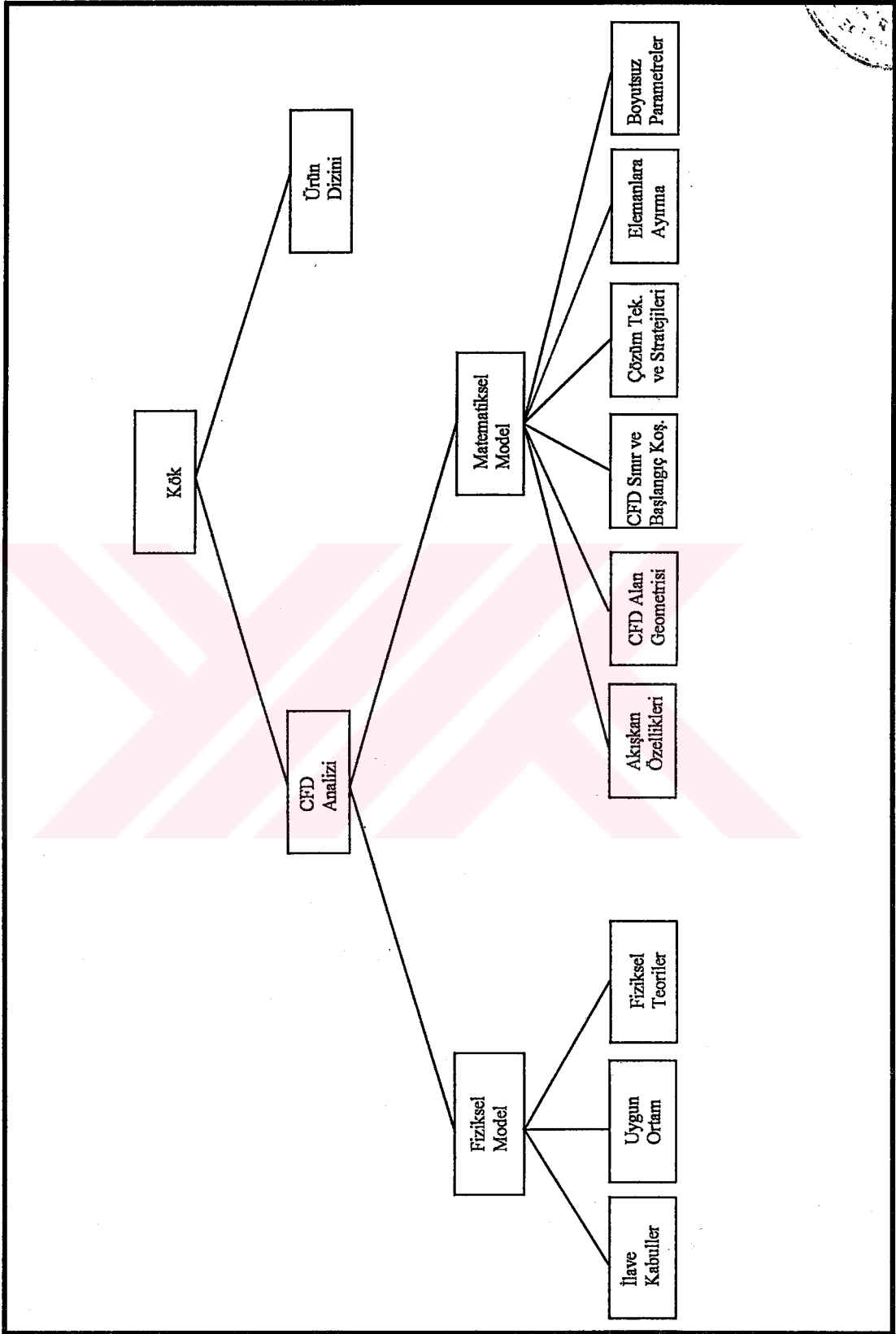


## 6.2 Yazılımın Hazırlanmasında Kullanılan Bilgisayar Ortamı

Bu bilgisayar yazılımı bir VAX 4500 iş istasyonu kullanılarak yapıldı. Bu makinada kullanılan işletim sistemi ise VAX/VMS V5.5 ve grafiksel kullanıcı arayüzü ise Xwindows altında çalışan Horses idi[66]. Ürün modelleme ortamının oluşturulmasında ise Leeds Üniversitesi tarafından geliştirilen Structure Editor (SE) kullanıldı[67]. SE özellikle Ürün Data Modellerinin oluşturulmasında, oluşturulan bu modellerin örneklenmesinde ve uygulamaları destekleyecek arayüz rutinlerinin yaratılmasında büyük kolaylık sağlamaktadır. Programlama dili olarak da SE programının yazıldığı dil olan ADA bilgisayar programlama dili kullanılmıştır [68,69].

## 6.3 CFD Analiz Data Modelinin Şematik Yapısı

Şekil 6.1'de akışkanlar mekaniği analizini desteklemek amacıyla hazırlanan data modelinin şematik yapısı görülmektedir. Bu data modeli, Structure Editor'ün grafik versiyonu olan Product Data Editor adlı program kullanılarak oluşturulmuştur. Şematik yapıdan da görüleceği gibi model iki ana kısımdan meydana gelmektedir. Bunlar *Fiziksel Model* kısmı ve *Matematiksel model* kısmıdır. Fiziksel model data yapısının altında ise *İlave Kabuller*, *Uygun Ortam*, ve *Fiziksel Teoriler* kısımları yer almaktadır. İlave Kabuller kısmında akışkan davranışları (örneğin akışkanın vizkoz veya vizkoz olmayan akışkan olması gibi) ve akışkan tipi (örneğin Newtonyen veya Newtonyen olmayan akışkan gibi) seçilmektedir. Uygun ortamın belirlenmesi kısmında ise, belirli bir problem için uygun olan ortamın söz konusudur (örneğin problemin ısı transferi problemi, akışkanlar mekaniği problemi olması gibi). Fiziksel Teoriler adlı kısımda ise uygun fiziksel teoriler (örneğin momentumun korunumu, enerjinin korunumu veya kütle korunumu gibi) seçilmektedir. Şekil 6.1'de görülen ikinci ana kısım ise Matematiksel Modeldir. Matematiksel model, dördüncü bölümde de açıklandığı gibi modeli oluşturmak için gerekli olan aktiviteleri içeren altı ayrı



Şekil 6.1 CFD data modelinin şematik yapısı



kısımdan meydana gelmektedir. Bu kısımlar, sırasıyla *Akışkan Özellikleri, CFD Alan Geometrisi, CFD Sınır ve Başlangıç Koşulları, Çözüm Teknikleri ve Stratejileri, Aracılız Alanının Süreksizleştirilmesi ve Boyutsuzlaştırma Parametreleri*'dir.

Yazılımın Akışkan Özellikleri kısmında akışkana ait fiziksel özellikler (yoğunluk ve viskozite gibi), ısıl özellikler (örneğin ısıl iletkenlik, ısı transfer katsayısı gibi) ve mekaniksel özellikler (örneğin kayma gerilmesi) belirlenmektedir.

CFD Alan Geometrisi kısmında ise, analiz edilecek alanın geometrik olarak tanımlanabilmesi için katı modelleme, yüzey modelleme tekniği veya eğri ve tepe noktaları aracılığıyla tanımlanması sağlanmaktadır. Geometrinin tanımlanması esnasında tepe noktalarının kullanılması halinde, geometriye ait anahtar noktalar hem kartezyen koordinat sisteminde hem de topolojik koordinatlarda tanımlanmaktadır.

CFD Sınır ve Başlangıç koşulları kısmında ise analiz edilecek olan alanın sınır koşulları örneğin hızın  $x$  ve  $y$  yönündeki bileşenlerinin değerleri ve başlangıç koşulları örneğin başlangıç anındaki hız gibi belirlenmektedir.

Çözüm teknikleri ve Stratejileri bölümünde ise, problemin tipi, örneğin lineer veya lineer olmayan sürekli akış, veya lineer veya lineer olmayan geçici akış gibi, çözüm teknikleri örneğin Quasi-Newton metodu ve iterasyon sayısı ve çözüm teknik veya teknikleri örneğin ardışık yerine koyma (successive substitution) ve iterasyon sayısı belirlenmektedir.

Süreksizleştirme yani sürekli bir alanın Sonlu elemanlar gibi bir yöntemle küçük elemanlara ayrılarak incelenmesi kısmında ise, hangi yöntemin kullanılacağı, kullanılacak olan eleman tipi belirlenmektedir.

Boyutsuzlaştırma parametrelerinin belirlendiği kısımda ise CFD analiz paketinin matematiksel denklemleri boyutsuz formda çözmek için gerekli olan, örneğin Reynolds Sayısı gibi parametreler belirlenmektedir.

## 6.4 CFD Data Modelinin Ürün Modelleme Ortamına Entegrasyonu İçin Arayüz Oluşturulması

Oluşturulan CFD analizi data modelinin örneklenmesi, yani belirli bir CFD problemi için gerekli bilgi ve dataların girilmesinden sonra bu bilgilerin Ürün Modelleme ortamında kullanılabilmesi ve CFD analiz paketinin ihtiyaç duyduğu bilgilerin belirli bir formda hazırlanıp bir dosyaya aktarılabilmesi için bir arayüz oluşturulması gerekmektedir. Bu arayüz, yine SE programının sağlamış olduğu arayüz için otomatik kod yazma imkanı ile gerçekleştirilmiştir[70,71]. Otomatik arayüz oluşturma işlemi sırasında, yaratılmış olan data modelindeki tüm noktalar taranarak bu noktalara örnekleme işlemi sırasında yerleştirilecek olan bilgilerin buradan çekilerek bir dosyaya aktarılması işlemini gerçekleştirecek bir program üretilir. Bu program, ADA bilgisayar programlama dili kullanılarak yapılmıştır. Fakat yukarıda sözü edilen yeni yaratılmış olan arayüz programı ham halde olup bu programın, uygun dataların istenilen formda bir dosyaya kaydedilmesi için yeniden düzenlenmesi gerekmektedir.

Bu arada, CFD analizi data modeli üzerinde gerçekleştirilecek her hangi düzeltme sonrası otomatik olarak üretilen arayüz programı geçerliliğini yitireceğinden data yapısı üzerinde yapılacak her düzeltmeden sonra otomatik kod üretme işleminin tekrar edilmesi gerekmektedir. Bu da doğal olarak yazılan bu kodların yeniden düzenlenmesi ve derlenmesi anlamına gelmektedir. Bu yüzden CFD analizi data modeli üzerinde yapılan değişiklikler ne kadar çok ise yapılması gereken düzeltmeler de o oranda artmaktadır.

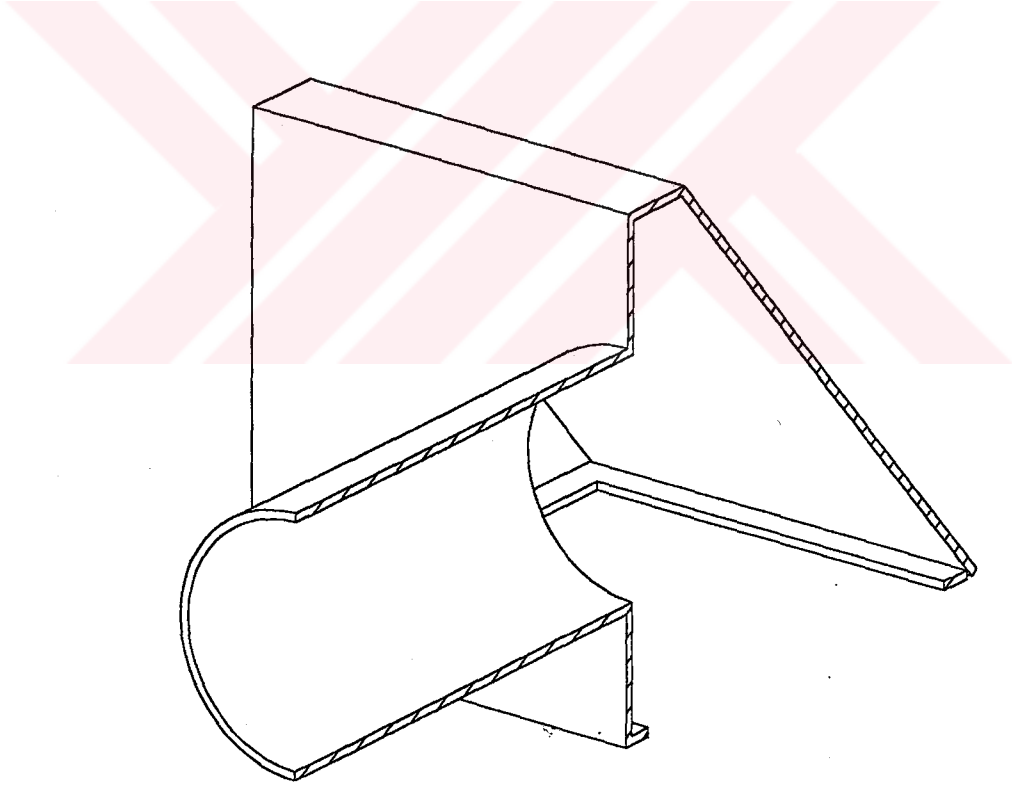
SE programı kullanılarak yapılan ve CFD analizi data modelinin Ürün Modelleme ortamına entegrasyonu için arayüz hazırlamak amacıyla otomatik olarak üretilen bilgisayar programından örnek bir kesit EK-C'de verilmiştir.





## 6.5 CFD Analizi Data Modelinin Örneklenmesi

Oluşturulan CFD analizi data modelinin örneklenmesi ve test edilmesi amacıyla bir gaz şofbeninin baca gazı çıkış kısmında bulunan parçanın içerisindeki gaz akışı analiz edildi. Gazlı şofbenin üst kısmında yer alan sözü edilen parça, ısı değiştirgecinden geçen yanmış gazın bacadan atılmasına yardımcı olan bir parçadır. Şekil 6.2'den de görüleceği üzere, yanmış gaz bu parçaya alt kısımdan girmekte ve sol taraftaki silindirik biçimli taraftan da dışarı atılmaktadır. Analizin yapılma amacı ise minimum enerji kaybı ve basınç düşmesiyle artık gazların dışarı atılmasını sağlayacak parçanın dizayn edilmesidir.



Şekil 6.2 Analizde kullanılan parçanın kesiti



Oluşturulacak olan 3 boyutlu analiz modeli, hem karmaşık olacağı hem de çözümü için daha fazla zaman gerekeceğinden sadece 2 boyutlu analiz modeli oluşturularak çözüme gidildi. Bu 2 boyutlu modelin çözümü ise, ticari bir CFD analiz paketi olan FIDAP V6.03 ile yapıldı. Analize temel teşkil eden girdi dosyasının bir örneği ise EK-D'de verilmiştir. Girdi dosyası, esas olarak iki ayrı kısımdan oluşmaktadır. FIMESH denilen ve analiz alanının süreksiz hale getirildiği ve sınır koşullarının belirlendiği birinci kısım ve FIPREP denilen ve problem spesifikasyonlarının, akışkan özelliklerinin, çözüm stratejilerinin, başlangıç koşullarının vb. belirlendiği ikinci kısım. Şimdi girdi dosyasında yer alan bu bilgilerin daha iyi anlaşılabilmesi için aşağıda bu dosyada yer alan bilgilerin incelenmesine geçilecektir.

#### 6.5.1 Girdi Dosyasındaki Bilgilerin Tanımlanması

Bu bölümde Ek-D'de bir örneği verilen, baca borusu CFD analizi için hazırlanan girdi dosyasındaki satırların ne anlama geldiği tanımlanmaya çalışılacaktır.

FIMESH (2-D,IMAX=5,JMAX=7)

Burada problemin 2 boyutlu olduğu, I ve J topolojik koordinatların maksimum değerlerinin ise 5 ve 7 olarak alınacağı ifade edilmektedir.

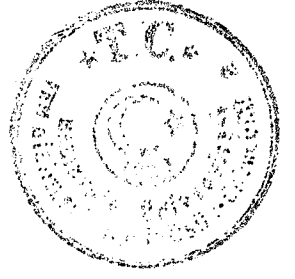
EXPI(DELTAS)

1 0 20 0 15

EXPJ(DELTAS)

1 0 10 0 15 0 13

Yukarıdaki bu dört satırla ise I ve J yönündeki topolojik koordinatlar genişletilmektedir. Komut satırındaki sıfırdan büyük olan sayılar, iki topolojik koordinat arasına ilave edilecek nokta sayısını göstermektedir.



#### POINT

1 3 1 1 1 0

2 5 1 1 2 0

Bu komut satırı ile de geometriye ait önemli noktaların koordinatları belirlenmektedir. (Bu noktalar genellikle geometrideki köşe noktalar olmaktadır.) Komutta yer alan ilk sayı, köşe noktasının numarasını, ardından gelen 3 sayı ise o noktanın topolojik koordinatlarını ve geri kalan son iki sayı da yine o noktanın fiziksel koordinatlarını ifade etmektedir.

#### LINE

2 3 .3 3

1 8 .3 3

11 12 .1 3

10 9 .1 3

Bu komut ile, geometri üzerinde belirlenen önemli noktaların birbirlerine bir doğru ile mi yoksa bir eğri ile mi bağlandığı tanımlanmaktadır. İlk iki rakam birbirlerine bağlanacak olan noktaları diğer iki nokta ise bu iki nokta arasındaki sonlu elemanların değişme derecesini göstermektedir.

#### SURFACE

1 5

11 9

Bu komut ile, analizi yapılacak olan bölgenin topolojik olarak dikdörtgen biçiminde tanımlanabilmesi için gerekli birbirine çarpaz pozisyonda olan noktalar belirlenmektedir.

#### MERGE

9 12 7 8



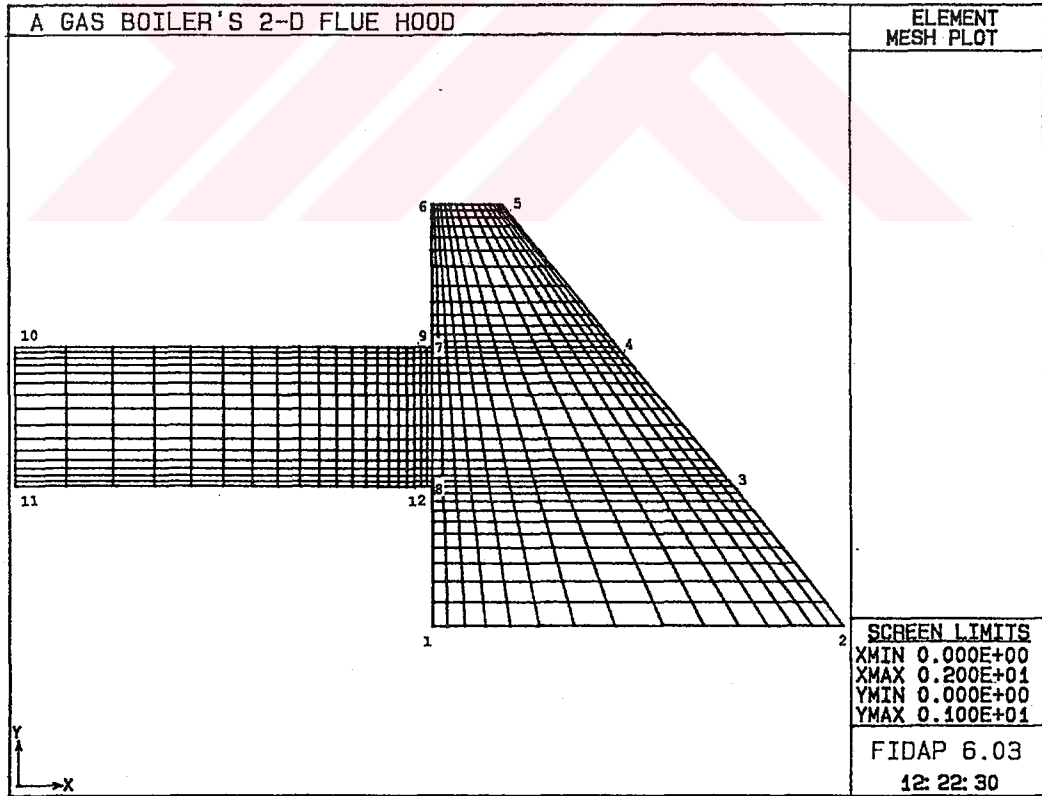
Analiz edilecek alanın topolojik olarak dikdörtgenlere ayrılması sırasında, herhangi bir şekilde üst üste gelen noktalar var ise, bu komut sayesinde bu noktalar birleştirilerek sanki tek bir nokta olarak işlem görmektedir.

ELEMENTS(QUADRILATERAL,NODES=4)

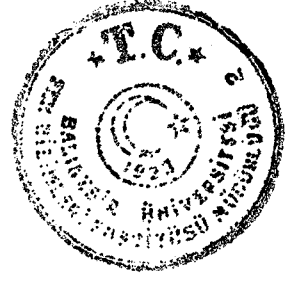
13 5

Bu komut ile de, analiz edilecek alanın süreksiz hale getirilmesi sırasında kullanılacak olan eleman tipi, eleman üzerinde dikkate alınacak nokta sayısı ve analiz edilecek olan alanın topolojik olarak dikdörtgeni oluşturacak çarpraz karşılıklı iki noktası belirlenmektedir.

Girdi dosyasında buraya kadar anlatılan komutlarla analizi yapılacak olan geometrik alan şekil 6.3'de görüldüğü gibi elemanlara ayrılmıştır.



Şekil 6.3 2 boyuta indirgenmiş örnek problemin sonlu elemanlara ayrılmış biçimi



BCNODE(UX)

1 2 0

2 5 0

5 6 0

Burada ise sınır koşulları belirlenmektedir. Örnekte, 1 ve 2, 2 ve 5, 5 ve 6 nolu noktalar arasındaki noktalarda hızın x bileşeninin "0" olduğu tanımlanmaktadır.

BCNODE(UY)

1 2 1.5

Bu komutla da 1 - 2 nolu noktalar arasında hızın y bileşeninin "1.5" olduğu ve hız profilinin ise parabolik olduğu tanımlanmaktadır.

BCNODE(UY)

5 6 0

5 - 6 nolu noktalar arasında hızın y bileşeninin "0" olduğu tanımlanmaktadır.

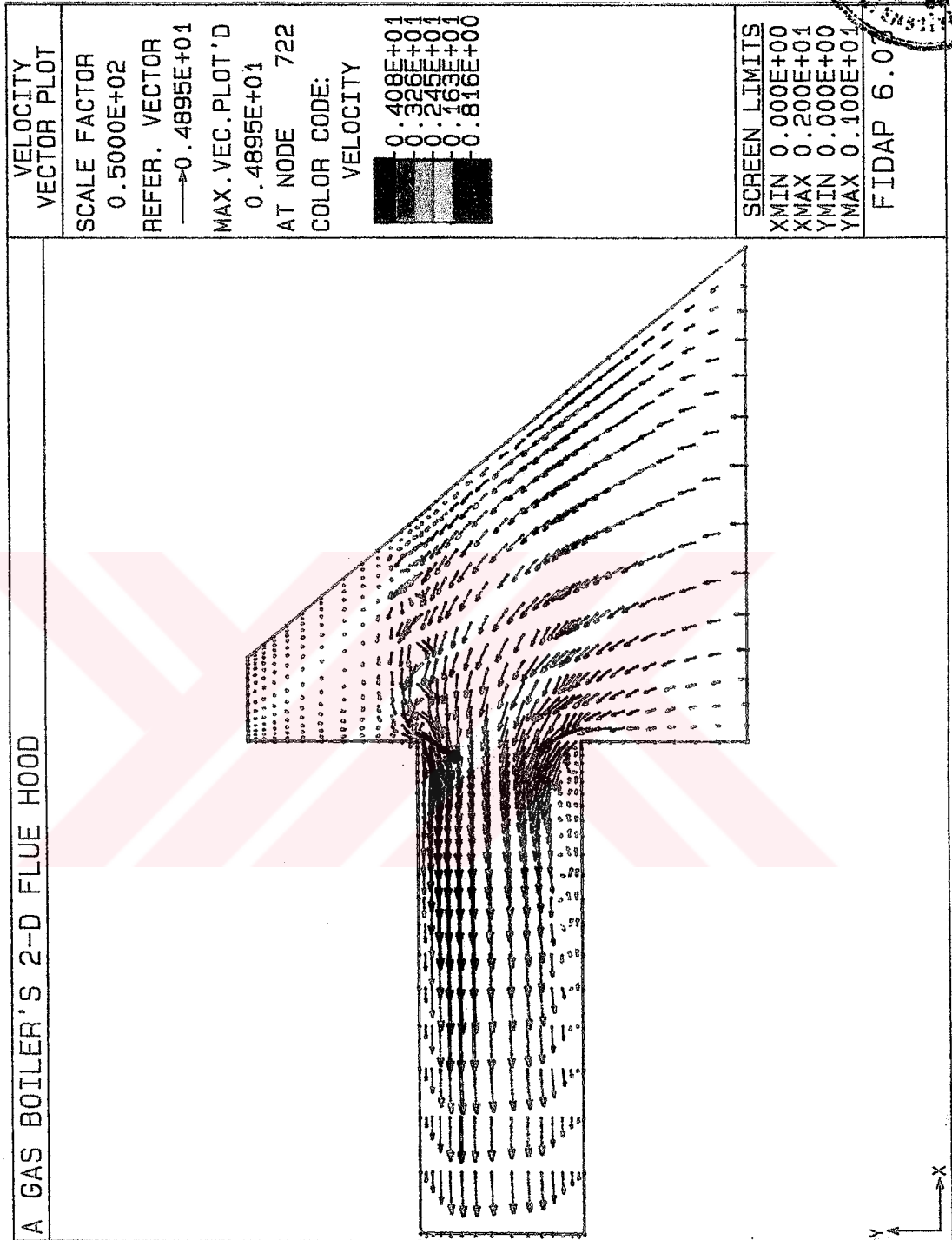
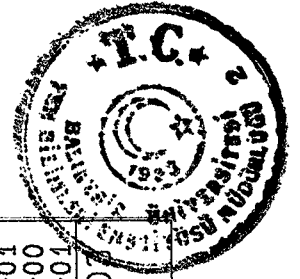
Girdi dosyasının ikinci kısmı olan FIPREP kısmına geçildiğinde ise:

PROBLEM(STEADY,NEWTONIAN,2D,MOMENTUM,ISOTHERMAL,FIXED,  
LAMINAR)

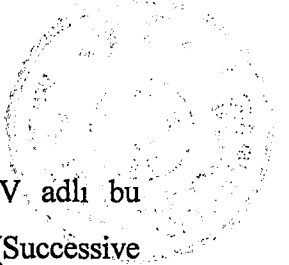
satırı görülecektir. Burada CFD problemi ile ilgili önemli tanımlar yapılmakta ve bilgiler verilmektedir. Örneğin analiz problemi için kullanılan bu satırda, problemin bir kararlı denge problemi olduğu, akışkanın Newton'yan bir akışkan olduğu, geometrinin iki boyutlu olarak alındığı, momentum denklemlerinin çözüleceği, problemin izoterm olduğu, geometrinin problemin çözümü sırasında sabit kaldığı ve problemin bir laminar akış problemi olduğu tanımlanmaktadır.

SOLUTION(Q:N = 10)

STRATEGY(S:S = 7)



Şekil 6.4 Analiz edilen parça içerisindeki gaz akışı

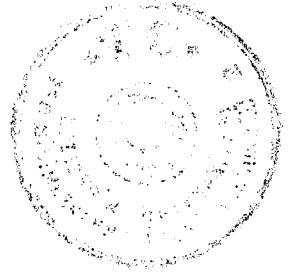


Çözüm teknik/tekniklerinin ve strateji/stratejilerinin belirlendiği FISOLV adlı bu kısımda, çözüm esnasında ilk yedi adıma kadar ardışık yedekleme (Successive Substitution) yönteminin kullanılacağı ve ardından da Quasi Newton yönteminin kullanılmasına geçileceği belirtilmektedir.

Problemin çözümünün ardından elde edilen analiz sonucunun bilgisayar ortamında değerlendirilmesi sonucu elde edilen ve bir şofbene ait baca parçasının içerisindeki akışkan akışını gösteren görüntü şekil 6.4'de verilmiştir.

### **6.6 Bölümsel Sonuç ve Değerlendirmeler**

Bu bölümde, çalışmanın dördüncü ve beşinci bölümlerinde oluşturulan aktivite ve bilgi modelleri temel alınarak, CFD analizini gerçekleştirebilmek için gerekli olan bilgileri kapsayacak bir CFD data modeli oluşturuldu. Oluşturulan bu CFD data modelinin test edilmesi amacıyla da örnek bir analiz problemi ele alındı ve bu analizi gerçekleştirmek için gerekli olan verilerin, oluşturulan CFD data model tarafından sağlanıp sağlanamayacağı kontrol edildi. Ayrıca yine bölüm içerisinde, analizde kullanılan girdi dosyasında yer alan bilgiler de açıklandı.



## 7. SONUÇ VE ÖNERİLER

### 7.1 GİRİŞ

Bu bölümde, yapılan çalışma sonucunda elde edilenler ve daha sonra yapılacak olan çalışmalar için öneriler yer almaktadır.

Bu tezde bahsedilen çalışmada, bilgisayar destekli akışkanlar mekaniği analizi (CFD) için gerekli olan bilgi ve aktiviteler araştırıldı. CFD analizi işlemi sırasında yer alan aktiviteler, SADT adı verilen bir modelleme tekniği kullanılarak ve analizi gerçekleştirebilmek için gerekli olan bilgiler ise EXPRESS-G denilen bilgi modelleme dili kullanılarak modellenmiştir. Yukarıda bahsedilen ve sırasıyla bölüm 4 ve bölüm 5'de detaylı olarak anlatılan bu modellerin ışığı altında, CFD analizi prosesini mühendislik tasarımı aşamasına entegre etmek amacıyla ürün modelleme denilen kavramdan faydalanılarak CFD data modeli denilen bir bilgisayar programı geliştirilmiş ve daha sonra bu yazılım bir örnek çalışma ile test edilerek, CFD analiz paket programı için gerekli olan dataların elde edilmesine çalışılmıştır.

### 7.2 Çalışmanın Sağladığı Katkılar

Bu çalışmanın sağladığı katkılar aşağıya maddeler halinde çıkarılmıştır;

- Bilgisayar destekli akışkanlar dinamiği (CFD) analiz prosesi esnasında yer alan aktiviteler belirlenip, bunların bir aktivite modelleme tekniği olan ve SADT olarak anılan bir teknik ile modellenerek analizdeki aktivitelerin daha iyi anlaşılabilmesi sağlandı.





- CFD analizi için gerekli olan bilgiler ve aktiviteler arasındaki bilgi akışları, EXPRESS bilgi modelleme dilinin grafik versiyonu olan EXPRESS-G kullanılarak modellendi.
- CFD analizini desteklemek ve gerekli bilgileri Ürün Modelleme ortamına sağlayabilmek amacıyla yapılan ve detayları beşinci bölümde verilen Bilgi Modeli baz alınarak, bilgisayar ortamında, ADA bilgisayar proglama dili ve ürün modeli yaratmak üzere özel olarak yazılmış olan Yapı Editörü (Structure Editor) kullanılarak bir CFD Ürün Data Modeli geliştirildi.
- Geliştirilen CFD Ürün Data Modelinin geçerliliğini sağlamak amacıyla da bir şofbenin parçası olan baca gazı borusu içerisindeki akışın analizi için gerekli olan bilgilerden örnek olarak yararlanıldı ve modelin gerekli bilgileri sağlayıp sağlayamayacağı kontrol edildi. Programdan elde edilen sonuçlar ise EK-E de verildi.

### 7.3 İleride Yapılacak Çalışmalar İçin Öneriler

Bu çalışmada, CFD analizinde yer alan aktivitelerin ve analizi yapabilmek için gerekli olan bilgilerin modellenmesi, CFD data modelinin geliştirilmesi ve bu modelin örneklenmesi sırasında aşağıdaki noktalar tespit edildi.

- Daha önce de belirtildiği üzere, CFD analizi sırasında yer alan aktiviteler, SADT denilen bir aktivite modelleme tekniği ile, bu analizi yapmak için gerekli olan bilgiler ise EXPRESS-G denilen bir bilgi modelleme tekniği ile modellendi.

Gerek aktivitelerin ve gerekse bilgilerin ayrı ayrı teknikler kullanılarak modellenmesi sonucu ortaya çıkan aktivite ve bilgi modellerinin birbirleriyle uyumlu olup olmadıklarını tespit etme imkanı bulunmamaktadır. Bu ise, kompleks olan ve birden çok sayfaya yayılan bir modelde, her iki model arasında uyumsuzluk olması ihtimalini arttırmaktadır.



Bu yüzden, hem aktiviteleri hem de o aktiviteleri gerçekleştirmek için gerekli olan bilgileri modelleme yeteneğine sahip birleştirilmiş bir modelleme tekniğine ihtiyaç bulunmaktadır.

- CFD analizinde kullanılan örneğe ait geometrik veriler, programa kullanıcı tarafından yüklendi. Bilindiği üzere son zamanlarda kullanılan CAD yazılım programlarından nesnelerin geometrik verilerinin IGES denilen veri değişim standardı formatında elde edilmesi mümkün olmaktadır. Ancak bu, CFD data model ile CAD yazılım programı arasında bir başka uygulama paketinin geliştirilmesine ihtiyaç duymaktadır.
- CFD analizi için kullanılacak olan geometrik alanın süreksizleştirilmesi bir başka deyişle elemanlara ayrılması işlemi sırasında yaygın olarak kullanım alanı bulan Sonlu Elemanlar Yöntemi (FEM) kullanıldı. Ancak geliştirilen CFD data modelinin daha genel olması isteniyorsa bu durumda diğer elemanlara ayırma yöntemleri olan Sınır Elemanlar Yöntemi, Sonlu Farklar Yöntemi ve Sonlu Hacimler Yöntemleri de modele dahil edilmelidir.
- Gerek CFD data modelinin geliştirilmesi sırasında ve gerekse bu modelin örneklenmesi sırasında sadece tek bir eleman göz önüne alındı. Fakat analizi yapılacak olan parça, birden fazla elemanın birbirine monte edilmesiyle meydana gelmişse, bu durumda bunun modele işlenmesi mümkün olmamaktadır. Bu konunun daha detaylı olarak araştırılması gerekmektedir.
- Halihazırda kullanılmakta olan CFD analiz paketleri, örneğin FIDAP, hesaplamaları minimum bilgisayar zamanı kullanarak yapabilmek için kullanıcıya sormaksızın bazı değerleri önceden kabul etmektedir. Analiz üzerindeki kontrolü tam olarak sağlayabilmek için CFD analiz programı tarafından yapılacak olan tüm bu kabullerin önceden bilinmesi ve bunların da oluşturulan CFD veri modeline dahil edilmesi gerekmektedir. Bu yüzden bu konunun derinlemesine incelenmesi ve bu değerlerin kontrol edilmesi gerekliliği vardır.

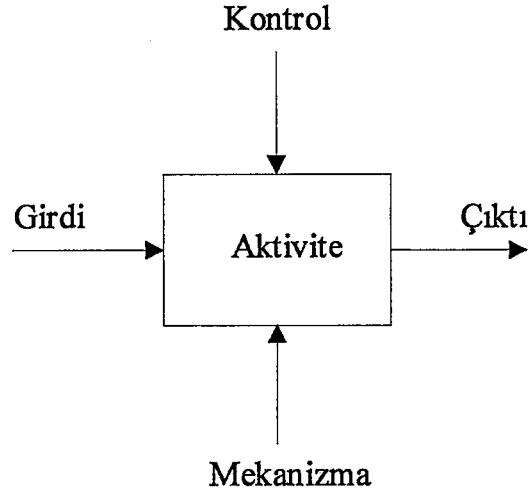
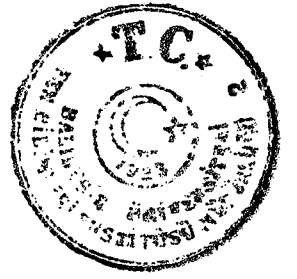


## EK-A : SADT AKTİVİTE MODELLEME TEKNİĞİ

SADT ve onun bilgisayar programı haline dönüştürülmüş versiyonu olan IDEF0, aktivitelerin modellenmesi için kullanılan bir grafiksel modelleme dilidir. SADT, İngilizce Structured Analysis and Design Technique (Yapısal Analiz ve Dizayn Tekniği) sözcüklerinin baş harflerinin alınmasıyla oluşturulmuş bir kısaltmadır. Bu tekniğin bilgisayar programı haline dönüştürülmüş versiyonu olan IDEF0 ise yine İngilizce Integrated Computer Aided Manufacturing Definition Method (Entegre edilmiş Bilgisayar Destekli İmalat Tanımlama Metodu) sözcüklerinden türetilmiş bir kısaltmadır.

SADT ilk olarak 1960'lı yıllarda yapısal programlama alanında köklü değişikliklerin yapıldığı sırada geliştirilmiştir. Bir sistemdeki aktiviteler üzerine yoğunlaşmış SADT modellerine “aktivite modelleri” adı verilmektedir [67]. Aktivite modelleri, bir sistemde mevcut aktiviteleri ardışık ve detaylı bir biçimde ifade ederken aynı zamanda bu aktiviteler arasındaki ilişkileri de ortaya koymaktadır.

Bir SADT modeli, bir sistemin eksiksiz bir tanımını sağlamakta ve yine SADT diyagramları denilen bir dizi diyagramdan oluşmaktadır. SADT diyagramı ise, dikdörtgen şeklindeki kutulardan ve ucu oklu çizgilerden meydana gelmektedir. Dikdörtgen biçimli kutu, bir fonksiyonu veya bir sistemin aktif kısmını ifade etmektedir. Bu dikdörtgen biçimli kutunun her bir kenarının ise özel bir anlamı vardır. Dikdörtgen kutunun sol tarafındaki kenarı girdiler, üst tarafındaki kenar kontroller, sağ tarafındaki kenar çıktılar ve alt tarafındaki kenar ise mekanizmalar için ayrılmıştır. (Bakınız Şekil A.1) Eğer Şekil A.1'i kelimelerle anlatmaya çalışacak olursak : Sisteme giren kontrollerin denetimi altında girdiler, mekanizmalar tarafından çıktılar haline dönüştürülmektedir.



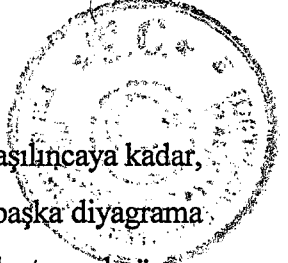
**Şekil A.1** Bir SADT kutusu ve okları

Bir aktivitenin sonucunda elde edilen bir çıktı, diğer bir aktiviteye girdi, kontrol veya mekanizma olarak girebilir. Aktiviteler arasındaki ilişkiyi sağlayan ucu oklu çizgiler ise, birbirleriyle birleşebilir veya dallara ayrılabilirler. Sistemdeki aktiviteleri temsil eden dikdörtgenler ise gerektiğinde detay diyagramlar üzerinde kendisini oluşturan alt aktivitelere ayrılabilirler.

Bir SADT diyagramı, birbirleriyle ucu oklu çizgiler aracılığıyla bağlanmış, üç ila altı arasında değişen sayıdaki dikdörtgen biçimli kutulardan oluşmaktadır. Bir sistemin yeterince tanımlanabilmesi için ise bu diyagramlardan en az birkaç tanesine ihtiyaç vardır. Oluşturulan bu diyagramlar bir araya getirilip birleştirildiğinde ise ortaya bir SADT modeli çıkmaktadır.

### **A.1 Diyagram Numaraları**

SADT modelleri, üstten-aşağıya doğru giden yapısal bir ayrıştırma işlemine tabi tutulurlar. Yapısal ayrıştırma işlemi sırasında, öncelikle modelin en üstünde yer alan tek kutucuk, içerisinde üç ila altı arasında aktiviteleri temsil eden kutuların bulunduğu bir başka diyagrama dönüştürülür. Ardından bu diyagram üzerinde yer alan üç ile altı

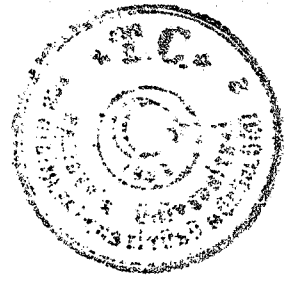


arasındaki aktiviteleri temsil eden kutular, istenilen detay derinliğine ulaşıncaya kadar, içerisinde yine üçten az altıdan çok olmamak şartıyla ayrıştırılarak bir başka diyagrama dönüştürülebilir. Her diyagrama verilecek olan isim, o diyagramı oluşturmak üzere ayrıştırılan aktivite kutusunun isminden alınmaktadır. Modelde yer alan her bir diyagram o modelin tümleşik bir parçasıdır. SADT modelleme tekniğinde her bir diyagrama “diyagram numaralama” denilen bir sistemle numara verilmektedir. Örneğin bir SADT modelinin en üst kısmında yer alan kontekst diyagramı, aşağıdaki gibi numaralanmaktadır: model adı veya kısaltması, bölü işareti (/), büyük harf A (buradaki A harfi aktiviteyi temsil etmektedir), tire işareti (-) ve 0 (A-0 gibi). Numaralama işlemini daha iyi izleyebilmek için şekil 4.1, 4.2, ve 4.3 de yer alan diyagramlara bakınız. Kontekst diyagramı ayrıştırıldığında oluşturulan diyagramın numaralanmasında ise yukarıda sıralanan formda sadece tire işareti kullanılmayacaktır. (A0 gibi). Yine bu diyagramın içerisinde yer alan aktivitelerin ayrıştırılması sonucu oluşturulacak diyagramlarda ise, ayrıştırılan aktivite kutusunun sağ alt köşesinde yer alan A1, A11, A111 şeklinde giden numaralar kullanılmaktadır. Bu sayede her diyagramın, SADT model hiyerarşisinde bulunduğu yer kolayca anlaşılabilir.

## A.2 ICOM Notasyonu

İyi bir yapısal analiz metodu, kullanıcıya tüm diyagramların birbirleriyle olan bağlantısını doğru olarak tanımlamaya yarayacak bir notasyon sistemine sahip olmalıdır. SADT modelleme tekniği ise bunu ICOM diye adlandırılan bir kodlama sistemi ile gerçekleştirmektedir. ICOM kısaltması, girdi, kontrol, çıktı ve mekanizma (Input, Control, Output, and Mechanism) sözcüklerinin İngilizce baş harflerinden oluşmaktadır. Bu kodlama sistemi sayesinde, SADT diyagramındaki bulunan ve sistemdeki aktiviteleri temsil eden dikdörtgenler herhangi bir karışıklığa ve şüpheye gerek kalmaksızın ayrıştırılmakta ve bir dikdörtgene giren ve çıkan okların tamamı ayrıştırmanın yapılacağı diyagrama aynen transfer edilmektedir.

SADT tekniği ile ilgili çok daha geniş bilgi ise [67] nolu kaynaktan elde edilebilir.



## **EK-B : EXPRESS-G BİLGİ MODELLEME DİLİ**

EXPRESS-G bir bilgi modelleme dili olan EXPRESS'in grafik versiyonudur [89]. Bir bilgi modelinin nesne, biçim, fonksiyon gibi şeylerin tanımını, bunlar üzerindeki kısıtlamaları, tanımlar arası ilişkileri ve ilişkiler üzerindeki kısıtlamaları içereceği kabul edilmektedir. EXPRESS-G modelleme dili kullanılarak yapılan bir bilgi modelindeki diyagramlar, bir takım grafik semboller kullanılarak oluşturulurlar. Bu diyagramlarda kullanılan üç tip sembol vardır. Bunlar sırasıyla;

1. **Tanımlama sembolleri** : Bu semboller bilgi modelinin temelini oluşturan kavram, düşünce gibi şeyleri ifade etmektedirler.
2. **İlişki sembolleri** : Bunlar ise tanımlama sembolleri arasındaki ilişkileri göstermek üzere kullanılırlar.
3. **Sayfa düzenleme sembolleri** : Bu semboller ise, bilgi modelindeki diyagramların birden fazla sayfaya yayılması durumunda kullanılırlar.

### **B.1 Tanımlama Sembolleri**

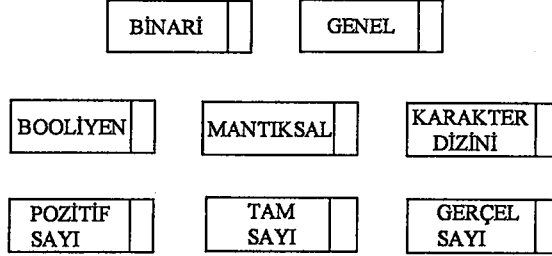
Tanımlama sembolü, tanımlanacak şeyin ismini içeren bir dikdörtgen kutucuktan oluşmaktadır. Tanımın tipi ise, dikdörtgen kutunun sınırlarını oluşturan çizginin tipine bağlıdır (düz çizgi, kesik çizgi gibi).

#### **B.1.1 EXPRESS'teki basit tip semboller**

EXPRESS modelleme dili, önceden belirlenmiş bazı basit tip sembollere sahiptir. Bunlar BİNARİ, BOOLİYEN, TAM SAYI , MANTIKSAL, POZİTİF SAYI, GERÇEL



SAYI, KARAKTER DİZİNİ ve GENEL olarak sınıflandırılabilirler. Bu semboller, kenarları düz çizgi ile çizilmiş ve sağ tarafta iki tane dikey çizgi bulunan dikdörtgen biçimli kutucuklarla gösterilmektedirler. Sembolün tipi ise kutucuğun içerisinde yer almaktadır (bakınız şekil B.1).



Şekil B.1 Basit tip semboller

### B.1.2 Tip Sembolleri

Tip sembolleri üç farklı türden oluşmaktadır. Bunlar; tanımlanmış data tipi, seçme tipi, ve liste tipi olarak sıralanabilir. Tanımlanmış data tipi sembolü, kesikli çizgi ile çizilmiş ve tipin ismini içeren bir dikdörtgenden meydana gelmektedir. Seçim tipi sembol ise, yine sınırları kesikli çizgi ile çizilmiş ve sol tarafında dikey çift çizgi bulunan ve seçimin ismini de içeren bir dikdörtgenden oluşmaktadır. Liste tipi sembol ise, yine sınırları kesikli çizgi ile çizilmiş olan ve bu kez sağ tarafında dikey çift çizgi bulunan dikdörtgenden oluşmaktadır. (Bakınız şekil B.2)

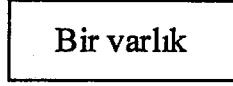
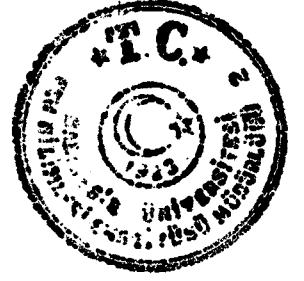


Şekil B.2 Tip tanımlama sembolleri

### B.1.3 Varlık Sembolü

Varlıkların simgelandığı bu sembol ise, sınırları düz çizgi ile çizilmiş ve içerisinde o varlığın isminin yer aldığı bir dikdörtgenden meydana gelmektedir (bakınız şekil B.3).

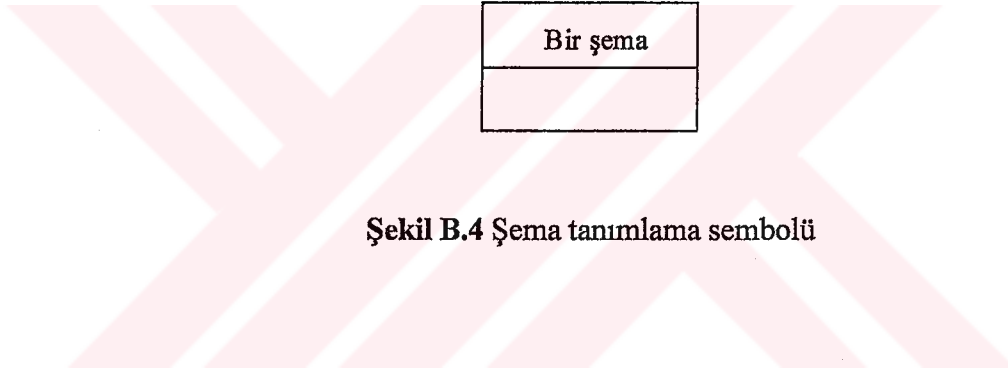




Şekil B.3 Varlık tanımlama sembolü

#### B.1.4 Şema Sembolü

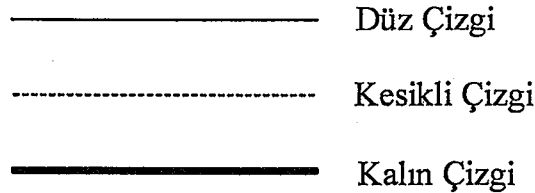
Şema sembolü, yatay olarak ikiye bölünmüş, sınırları düz çizgi ile çizilmiş ve şemanın adının bölünmüş olan dikdörtgenin üst kısmında yazıldığı bir dikdörtgenden meydana gelmektedir. İkiye bölünmüş dikdörtgenin alt kısmına herhangi bir şey yazılmıyabilir. (Bakınız şekil B.4).



Şekil B.4 Şema tanımlama sembolü

#### B.1.5 İlişki Sembolleri

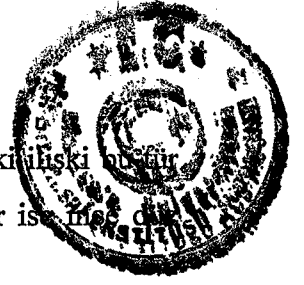
Tanımlama sembolleri birbirlerine farklı tipteki çizgilerle bağlanabilmektedir (bakınız şekil B.5).



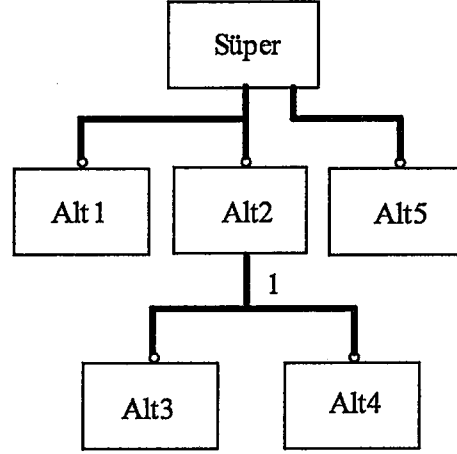
Şekil B.5 İlişkileri İfade Etmekte Kullanılan Çizgi Tipleri

Bir nesnenin isteğe bağlı olan herhangi bir özelliği ile arasındaki ilişki, kesikli çizgi ile ifade edilmektedir. Ağaç diyagramındaki ilişkiler ise, kalın düz çizgi ile





gösterilmektedir (bakınız şekil B.6). Örneğin süpertip ile alttip arasındaki ilişki bütür bir çizgi ile ifade edilmektedir. Bunların dışındaki diğer bütün ilişkiler ise ince bir çizgi ile gösterilir.



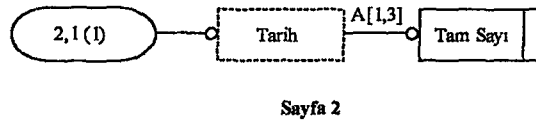
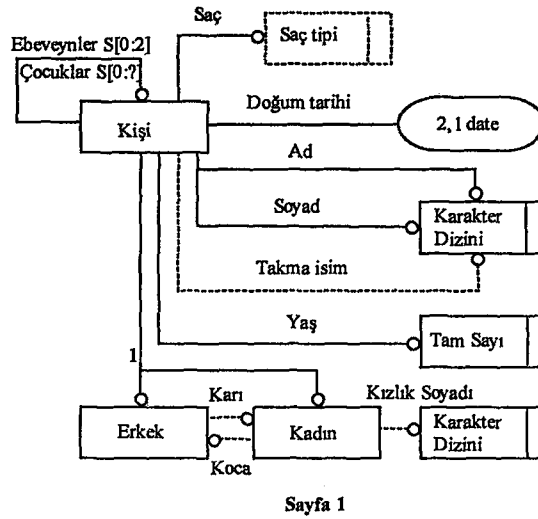
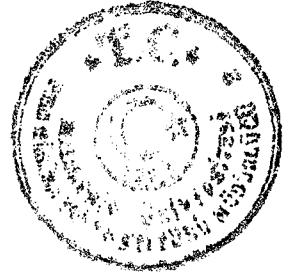
Şekil B.6 Ağaç Diyagramındaki İlişkiler

## B.2 Sayfa Düzenleme Sembolleri

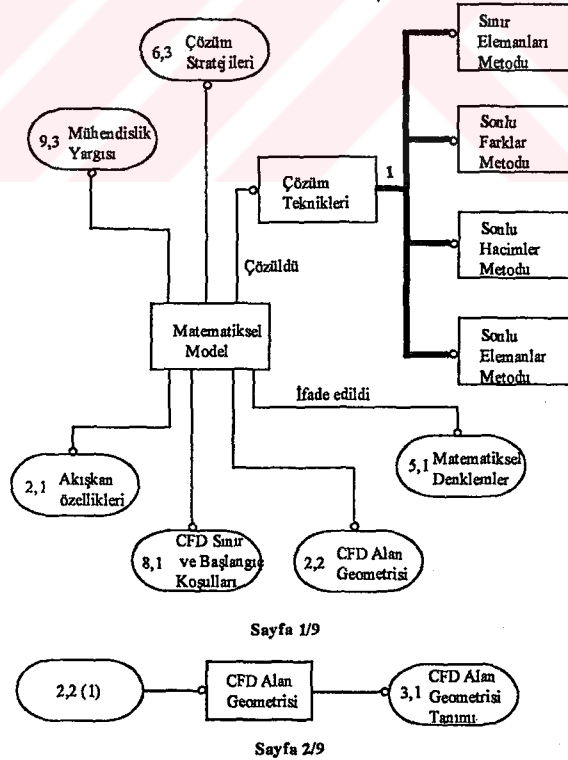
Bilgi modelinin grafiksel biçimde ifadesi genellikle birden fazla sayfaya yayılmakta ve bu da modelin takibini zorlaştırmaktadır. Bu yüzden modeldeki diyagramların yer aldığı her sayfa belirli bir formatta numaralanmalıdır.

### B.2.1 Bir Sayfayı Referans Verme

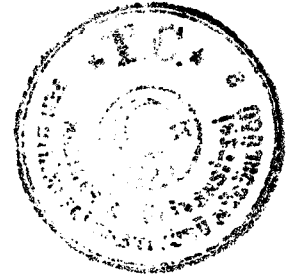
Eğer farklı sayfalarda bulunan tanımlar arasında herhangi bir ilişki varsa, bu durumda, ilişki çizgisi, köşeleri yuvarlatılmış dikdörtgen kutuya iliştilmelidir. Köşeleri yuvarlatılmış olan bu kutu, referans verilen sayfanın numarasını, referans numarasını ve tanımın adını içermektedir. Referans gösterilen sayfada da bir eşi bulunan köşeleri yuvarlatılmış dikdörtgen kutu, yukarıda bahsedilenlere ek olarak referansın verildiği sayfanın numarasını da parantez içerisine alınmış olarak bulundurabilir (bakınız Şekil B.7 ve B.8).



Şekil B.7 EXPRESS-G bilgi modelleme tekniğinde bir sayfayı referans verme



Şekil B.8 EXPRESS-G ile oluşturulan CFD bilgi modelinde bir başka sayfayı referans gösterme



**EK-C : BİR GAZ ŞOFBENİNE AİT BACA PARÇASININ FIDAP CFD  
ANALİZ PROGRAMI İLE ANALİZİ İÇİN HAZIRLANAN GİRĐİ  
DOSYASI**

TITLE

A GAS BOILER'S 2-D FLUE HOOD

\*FIMESH(2-D,IMAX=5,JMAX=7)

EXPI(DELTAS)

1 0 20 0 15

EXPJ(DELTAS)

1 0 10 0 15 0 13

POINT

1 3 1 1 1 0

2 5 1 1 2 0

3 5 3 1 1.73 .33

4 5 5 1 1.45 .66

5 5 7 1 1.17 1

6 3 7 1 1 1

7 3 5 1 1 .66

8 3 3 1 1 .33

9 3 5 1 1 .66

10 1 5 1 0 .66

11 1 3 1 0 .33

12 3 3 1 1 .33

13 1 1 1 0 0

LINE

1 2 3.3 3 3.3 .5

2 3.3 3



34.34.3.5

45.34.3.5

65.33.33.5

18.33

87.34.3.5

76.34.3.5

83.33.33.5

74.33.33.5

11.12.13

109.13

129.34.3.5

11.10.34.3.5

SURFACE

15

119

MERGE

9.12.7.8

ELEMENTS(QUADRILATERAL,NODES=4)

135

BCNODE(UX)

120

250

560

670

810

9100

11.12.0

BCNODE(UY,PARABOLIC)

12.1.5

BCNODE(UY)

560

250



670

1090

10110

11120

810

END

\*FIPREP

PROBLEM(STEADY,NEWTONIAN,2-D,MOMENTUM, ISOTHERMAL, FIXED,  
LAMINAR)

SOLUTION(Q.N.=10)

STRATEGY(S.S.=7)

EXECUTION(NEWJOB)

DATAPRINT(CONTROL)

POSTPROCESS(ALL)

NODES(CARTESIAN,FIMESH)

DENSITY(CONSTANT=1)

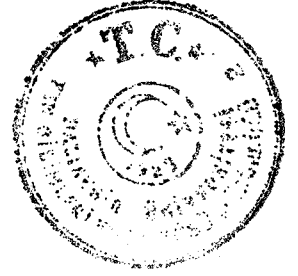
VISCOSITY(CONSTANT=.001)

ELEMENTS(QUADRILATERAL, NODES=4,FLUID,FIMESH)

ICNODE(VELOCITY,STOKES)

END

\*END



**EK-D : CFD ANALİZ VERİLERİNİ ÜRÜN MODELLEME ORTAMINDAN  
MANİPÜLE ETMEK İÇİN KULLANILAN YAZILIMDAN ÖRNEK TEŞKİL  
ETMESİ AMACIYLA ALINMIŞ PROGRAM KESİTİ<sup>1</sup>**

---

---

```
-- CFD_ANALYSIS.ADA
```

---

---

```
with gmp_io; use gmp_io;
with gmp ; use gmp ;
with sundries ; use sundries ;
with linked_list ;
with instance_definitions ; use instance_definitions ;
with meta_definitions ; use meta_definitions ;
with create_instances ; use create_instances ;
with se_utilities ; use se_utilities ;
with screen_text_io ; use screen_text_io ;
```

---

---

```
with physical_theories ; use physical_theories ;
with relevant_env ; use relevant_env;
with additional_assumptions ; use additional_assumptions ;
with n_dimen_parameters ;use n_dimen_parameters;
with discretization ;use discretization;
with env_pro_build ;use env_pro_build;
with domain_geometry ;use domain_geometry;
with boundary_cond ;use boundary_cond;
with solution_tech ;use solution_tech;
```

---

---

```
package body cfd_analysis is
```

```
procedure create ( x : name_atom_typ ; i : node_inst ) ;
-- creates a node_inst from x
```

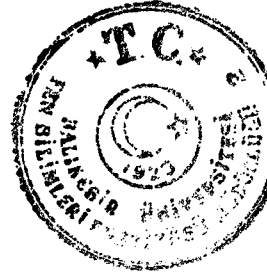
```
procedure create ( x : physical_model_typ ; i : node_inst ) ;
-- creates a node_inst from x
```

```
procedure create ( x : mathematical_model_typ ; i : node_inst ) ;
-- creates a node_inst from x
```

```
procedure create ( x : root_typ ; i : node_inst ) is
```

---

<sup>1</sup> Kaplıyacağı sayfa-sayısı göz önüne alındığında programın tamamını burada vermek uygun olmayacaktır.



```
-- creates a node_inst from x
  i1 , i2 , i3 , i4 : node_inst ;
begin
  i1 := i ;
  i2 := i1.sons.hd ; -- title_of_problem
  i3 := i1.sons.tl.hd ; -- physical_model
  i4 := i1.sons.tl.tl.hd ; -- mathematical_model
  if x /= null then
    create ( x.the_title_of_problem , i2 ) ;
    create ( x.the_physical_model , i3 ) ;
    create ( x.the_mathematical_model , i4 ) ;
  end if ;
end ;

procedure create ( x : name_atom_typ ; i : node_inst ) is
-- creates a node_inst from x
  i1 : node_inst ;
begin
  i1 := i ;
  i.atom_val := new atom_rec ( name_atom ) ;
  i.atom_val.name_val := x ;
end ;

procedure create ( x : physical_model_typ ; i : node_inst ) is
-- creates a node_inst from x
  i1 , i2 , i3 , i4 : node_inst ;
begin
  i1 := i ;
  i2 := i1.sons.hd ; -- additional_assumptions
  i3 := i1.sons.tl.hd ; -- relevant_environment
  i4 := i1.sons.tl.tl.hd ; -- physical_theories
  if x /= null then
    create ( x.the_additional_assumptions , i2 ) ;
    create ( x.the_relevant_environment , i3 ) ;
    create ( x.the_physical_theories , i4 ) ;
  end if ;
end ;

procedure create ( x : mathematical_model_typ ; i : node_inst ) is
-- creates a node_inst from x
  i1 , i2 , i3 , i4 , i5 , i6 , i7 : node_inst ;
begin
  i1 := i ;
  i2 := i1.sons.hd ; -- non_dimensionalization_parameters
  i3 := i1.sons.tl.hd ; -- problem_discretization_m
  i4 := i1.sons.tl.tl.hd ; -- environmental_properties
  i5 := i1.sons.tl.tl.tl.hd ; -- cfd_domain_geometry
  i6 := i1.sons.tl.tl.tl.tl.hd ; -- cfd_boundary_initial_c
```



```
i7 := i1.sons.tl.tl.tl.tl.tl.hd ; sol_tech_and_strategies
if x /= null then
  create ( x.the_non_dimensionalization_parameters,i2 ) ;
  create ( x.the_problem_discretization_m , i3 ) ;
  create ( x.the_environmental_properties , i4 ) ;
  create ( x.the_cfd_domain_geometry , i5 ) ;
  create ( x.the_cfd_boundary_initial_c , i6 ) ;
  create ( x.the_solution_tech_and_strategies , i7 ) ;
end if ;
end ;
```

---

```
function walk2 ( i : instance_definitions.node_inst)
return root_tp is
```

```
subtype root_tp is root_tp;
subtype name_atom_tp is name_atom_tp;
subtype physical_model_tp is physical_model_tp;
```

```
subtype additional_assumptions_tp is additional_assumptions_tp;
additional_assumptions_create : additional_assumptions_tp ;
```

```
subtype relevant_environment_tp is relevant_environment_tp;
relevant_env_create : relevant_environment_tp ;
```

```
subtype physical_theories_tp is physical_theories_tp;
physical_theories_create : physical_theories_tp ;
```

```
subtype mathematical_model_tp is mathematical_model_tp;
```

```
subtype non_dimensionalization_parameters_tp is
non_dimensionalization_parameters_tp;
n_dimen_parameters_create : non_dimensionalization_parameters_
tp ;
```

```
subtype problem_discretization_m_tp is
problem_discretization_m_tp;
discretization_create : problem_discretization_m_tp ;
```

```
subtype environmental_properties_tp is
environmental_properties_tp;
env_pro_build_create : environmental_properties_tp ;
```

```
subtype cfd_domain_geometry_tp is cfd_domain_geometry_tp;
domain_geometry_create : cfd_domain_geometry_tp ;
```

```
subtype cfd_boundary_initial_c_tp is cfd_boundary_initial_c_tp;
boundary_cond_create : cfd_boundary_initial_c_tp ;
```





```
subtype solution_tech_and_strategies_tp is
solution_tech_and_strategies_tp;
solution_tech_create : solution_tech_and_strategies_tp ;

no_name : sundries.identifier := sundries.blank_identifier ;
active_node : node_inst;

t0 : root_tp;
t1 : name_atom_tp;
t2 : physical_model_tp;
t4 : additional_assumptions_tp;
t5 : relevant_environment_tp;
t6 : physical_theories_tp;
t3 : mathematical_model_tp;
t7 : non_dimensionalization_parameters_tp;
t8 : problem_discretization_m_tp;
t9 : environmental_properties_tp;
t10 : cfd_domain_geometry_tp;
t11 : cfd_boundary_initial_c_tp;
t12 : solution_tech_and_strategies_tp;
i0,i1,i2,i3,i4,i5,i6,i7,i8,i9,
i10,i11,i12,i13 : node_inst;

function make_physical_model_tp(t1 : additional_assumptions_tp;
                               t2 : relevant_environment_tp;
                               t3 : physical_theories_tp )
return physical_model_tp is
begin
return new physical_model_rec'
( the_additional_assumptions =>t1 ,
  the_relevant_environment   =>t2 ,
  the_physical_theories      =>t3 );
end ;

function make_mathematical_model_tp(t1 :
non_dimensionalization_parameters_tp;
t2 : problem_discretization_m_tp;
t3 : environmental_properties_tp;
t4 : cfd_domain_geometry_tp;
t5 : cfd_boundary_initial_c_tp;
t6 : solution_tech_and_strategies_tp )
return mathematical_model_tp is
begin
return new mathematical_model_rec'
( the_non_dimensionalization_parameters => t1 ,
  the_problem_discretization_m         => t2 ,
  the_environmental_properties         => t3 ,
```



```
        the_cfd_domain_geometry      => t4 ,
        the_cfd_boundary_initial_c   => t5 ,
        the_solution_tech_and_strategies => t6 );
end ;

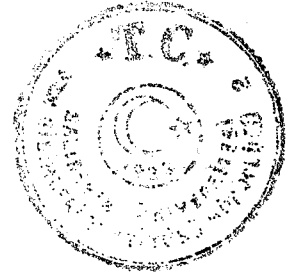
function make_root_tp( t1 : name_atom_tp;
                      t2 : physical_model_tp;
                      t3 : mathematical_model_tp )
return root_tp is
begin
return new root_rec'
( the_title_of_problem      => t1 ,
  the_physical_model        => t2 ,
  the_mathematical_model    => t3 );
end ;

function make_name_atom( x : identifier)
return name_atom_tp is
begin
return x ;
end ;

procedure analyse_root( i : node_inst;
                       title_of_problem1 : out node_inst ;
                       physical_model2 : out node_inst ;
                       mathematical_model3 : out node_inst ) is
wl : inst_list := i.sons;
begin
title_of_problem1 := wl.hd;
wl := wl.tl;
physical_model2 := wl.hd;
wl := wl.tl;
mathematical_model3 := wl.hd;
end;

procedure analyse_physical_model( i : node_inst;
                                  additional_assumptions1 : out node_inst ;
                                  relevant_environment2 : out node_inst ;
                                  physical_theories3 : out node_inst ) is
wl : inst_list := i.sons;
begin
additional_assumptions1 := wl.hd;
wl := wl.tl;
relevant_environment2 := wl.hd;
wl := wl.tl;
physical_theories3 := wl.hd;
end;

procedure analyse_mathematical_model( i : node_inst;
                                       non_dimensionalization_parameters1 : out node_inst ;
```

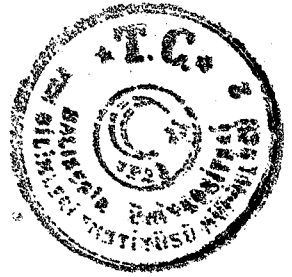


```
    problem_discretization_m2 : out node_inst ;
    environmental_properties3 : out node_inst ;
    cfd_domain_geometry4 : out node_inst ;
    cfd_boundary_initial_c5 : out node_inst ;
    solution_tech_and_strategies6 : out node_inst ) is
  wl : inst_list := i.sons;
begin
  non_dimensionalization_parameters1 := wl.hd;
  wl := wl.tl;
  problem_discretization_m2 := wl.hd;
  wl := wl.tl;
  environmental_properties3 := wl.hd;
  wl := wl.tl;
  cfd_domain_geometry4 := wl.hd;
  wl := wl.tl;
  cfd_boundary_initial_c5 := wl.hd;
  wl := wl.tl;
  solution_tech_and_strategies6 := wl.hd;
end;

function walk3(i : node_inst) return name_atom_tp is
  t0 : name_atom_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  if i0.atom_val = null
  then
    t0 := no_name;
  else
    t0 := make_name_atom(i0.atom_val.name_val);
  end if;
  return t0;
end walk3;

function walk4(i:node_inst) return additional_assumptions_tp is
  t0 : additional_assumptions_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := additional_assumptions.walk2(i);
  return t0;
end walk4;

function walk5(i : node_inst) return relevant_environment_tp is
  t0 : relevant_environment_tp;
  i0,i1 : node_inst;
begin
```



```
i0 := i;
  active_node := i0;
  t0 := relevant_env.walk2(i);
  return t0;
end walk5;
```

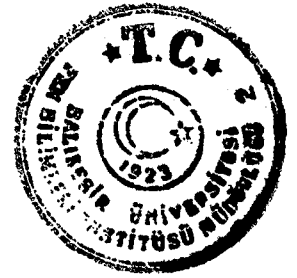
```
function walk6(i : node_inst) return physical_theories_tp is
  t0 : physical_theories_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := physical_theories.walk2(i);
  return t0;
end walk6;
```

```
function walk7(i : node_inst) return non_dimensionalization_
parameters_tp is
  t0 : non_dimensionalization_parameters_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := n_dimen_parameters.walk2(i);
  return t0;
end walk7;
```

```
function walk8(i : node_inst) return problem_discretization_
m_tp is
  t0 : problem_discretization_m_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := discretization.walk2(i);
  return t0;
end walk8;
```

```
function walk9(i : node_inst) return environmental_properties_
tp is
  t0 : environmental_properties_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := env_pro_build.walk2(i);
  return t0;
end walk9;
```

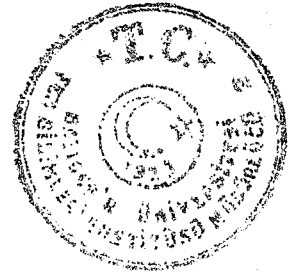
```
function walk10(i : node_inst) return cfd_domain_geometry_tp is
```



```
t0 : cfd_domain_geometry_tp;
i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := domain_geometry.walk2(i);
  return t0;
end walk10;
function walk11(i : node_inst) return cfd_boundary_initial_c_
tp is
  t0 : cfd_boundary_initial_c_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := boundary_cond.walk2(i);
  return t0;
end walk11;

function walk12(i : node_inst) return solution_tech_and_
strategies_tp is
  t0 : solution_tech_and_strategies_tp;
  i0,i1 : node_inst;
begin
  i0 := i;
  active_node := i0;
  t0 := solution_tech.walk2(i);
  return t0;
end walk12;

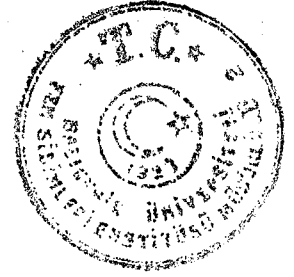
begin
  i0 := i;
  analyse_root(i0,
    title_of_problem1 => i1,
    physical_model2 => i2,
    mathematical_model3 => i3
  );
  t1 := walk3(i1);
  analyse_physical_model(i2,
    additional_assumptions1 => i4,
    relevant_environment2 => i5,
    physical_theories3 => i6
  );
  t4 := walk4(i4);
  t5 := walk5(i5);
  t6 := walk6(i6);
  active_node := i2;
  t2 := make_physical_model_tp(
```



```
t4,
t5,
t6 );
analyse_mathematical_model(i3,
  non_dimensionalization_parameters1 => i7,
  problem_discretization_m2 => i8,
  environmental_properties3 => i9,
  cfd_domain_geometry4 => i10,
  cfd_boundary_initial_c5 => i11,
  solution_tech_and_strategies6 => i12
);
t7 := walk7(i7);
t8 := walk8(i8);
t9 := walk9(i9);
t10 := walk10(i10);
t11 := walk11(i11);
t12 := walk12(i12);
active_node := i3;
t3 := make_mathematical_model_tp(
  t7,
  t8,
  t9,
  t10,
  t11,
  t12 );
active_node := i0;
t0 := make_root_tp(
  t1,
  t2,
  t3 );
return t0;
end walk2;
```

---

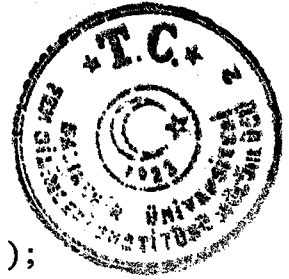
```
procedure put ( x : physical_model_typ ; outf : gmp_io.file_
type ) is
begin
  if x /= null then
    gmp_io.put (outf, " ADDITIONAL ASSUMPTIONS : " );
    gmp_io.new_line(outf);
    put ( x.the_additional_assumptions ,outf) ;
    gmp_io.new_line(outf);
    gmp_io.put (outf, " RELEVANT ENVIRONMENT : " );
    gmp_io.new_line(outf);
```



```
put ( x.the_relevant_environment ,outf) ;
gmp_io.new_line(outf);
gmp_io.put (outf, " PHYSICAL THEORIES : " );
gmp_io.new_line(outf);
put ( x.the_physical_theories ,outf) ;
gmp_io.new_line(outf);
end if ;
end ;

procedure put ( x : mathematical_model_typ ; outf :
gmp_io.file_type ) is
begin
  if x /= null then
    gmp_io.put (outf,"NON DIMENSIONALIZATION PARAMETERS : ");
    gmp_io.new_line(outf);
    put ( x.the_non_dimensionalization_parameters ,outf) ;
    gmp_io.new_line(outf);
    gmp_io.put (outf,"PROBLEM DISCRETIZATION METHODS : " );
    gmp_io.new_line(outf);
    put ( x.the_problem_discretization_m ,outf) ;
    gmp_io.new_line(outf);
    gmp_io.put (outf, " ENVIRONMENTAL PROPERTIES : " );
    gmp_io.new_line(outf);
    put ( x.the_environmental_properties ,outf) ;
    gmp_io.new_line(outf);
    gmp_io.put (outf, " CFD DOMAIN GEOMETRY : " );
    gmp_io.new_line(outf);
    put ( x.the_cfd_domain_geometry ,outf) ;
    gmp_io.new_line(outf);
    gmp_io.put (outf,"CFD BOUNDARY & INITIAL CONDITIONS : ");
    gmp_io.new_line(outf);
    put ( x.the_cfd_boundary_initial_c ,outf) ;
    gmp_io.new_line(outf);
    gmp_io.put (outf,"SOLUTION TECHNIQUES & STRATEGIES : " );
    gmp_io.new_line(outf);
    put ( x.the_solution_tech_and_strategies ,outf) ;
    gmp_io.new_line(outf);
  end if ;
end ;

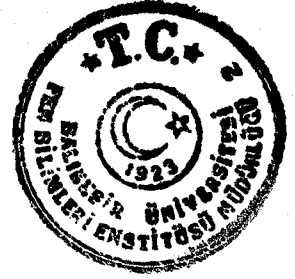
procedure put ( x : root_typ ) is
success : boolean ;
outf : gmp_io.file_type ;
begin
gmp_io.create (outf,"cfd_input","dat",success,true);
  if x /= null then
    gmp_io.put (outf, "THE TITLE OF PROBLEM : " );
    gmp_io.put (outf,spelling( x.the_title_of_problem ).all);
```



```
gmp_io.new_line(outf);
gmp_io.put (outf, "THE PHYSICAL MODEL :");
gmp_io.new_line(outf);
gmp_io.put (outf, "=====");
gmp_io.new_line(outf);
put ( x.the_physical_model ,outf);
gmp_io.new_line(outf);
gmp_io.put (outf, "THE MATHEMATICAL MODEL :");
gmp_io.new_line(outf);
gmp_io.put (outf, "=====");
gmp_io.new_line(outf);
put ( x.the_mathematical_model ,outf);
gmp_io.new_line(outf);
end if;
gmp_io.close(outf,success);
end;

end cfd_analysis ;
```





**EK-E :OLUŞTURULAN CFD ANALİZİ DATA MODELİNİN ÖRNEK BİR UYGULAMA İÇİN ÇALIŞTIRILMASI SONUCU ELDE EDİLEN ÇIKTILAR :**

THE TITLE OF PROBLEM : Flue Hood Analysis  
THE PHYSICAL MODEL :

**ADDITIONAL ASSUMPTIONS :**

fluid\_behaviour :  
viscous\_fluid :  
    newtonian\_fluid : selected  
    non\_newtonian\_fluid :-  
  
inviscid\_fluid :-

**RELEVANT ENVIRONMENT :**

fluid\_flow : selected  
heat\_transfer :-  
chemical\_reaction :-  
crystalization :-

**PHYSICAL THEORIES :**

conservation\_of\_momentum : selected  
conservation\_of\_energy :-  
conservation\_of\_mass :-

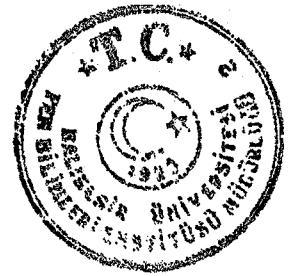
**THE MATHEMATICAL MODEL :**

**NON DIMENSIONALIZATION PARAMETERS :**

the\_reynolds\_number (Re) : 150.00  
the\_prandtl\_number (Pr) : 0.00  
the\_schmidt\_number (Sc) : 0.00  
the\_lewis\_number (Le) : 0.00  
the\_peclet\_number (Pe) : 0.00  
the\_rayleigh\_number (Ra) : 0.00  
the\_grashof\_number (Gr) : 0.00  
the\_capillary\_number (Ca) : 0.00

**PROBLEM DISCRETIZATION METHODS :**

the\_finite\_element\_method :  
two\_dimensional\_element :



quadrilateral :  
the number\_of\_nodal\_points\_of\_quadrilateral :  
four\_nodes\_quadrilateral : selected  
eight\_node\_quadrilateral :-  
nine\_node\_quadrilateral :-

triangle :  
the number\_of\_nodal\_points\_of\_triangle :  
three\_nodes\_triangle :-  
six\_nodes\_triangle :-  
seven\_nodes\_triangle :-

three\_dimensional\_element :  
brick :  
the number\_of\_nodal\_points\_of\_brick :  
eight\_nodes\_brick :-  
twentyseven\_nodes\_brick :-

tetrahedron :  
the number\_of\_nodal\_points\_of\_tetrahedron :  
four\_nodes\_tetrahedron :-  
ten\_nodes\_tetrahedron :-

wedge :  
number\_of\_nodal\_points\_of\_wedge :  
eight\_nodes\_wedge :-  
eighteen\_nodes\_wedge :-

#### FLUID PROPERTIES :

the physical\_properties :  
surface\_tension :  
surface\_tension\_value : 0.0000  
the unit :-

molecular\_weight :  
molecular\_weight\_value : 0.0000  
the unit :-

density :  
the density\_value : 1.0000



```

    the unit                :-
viscosity      :
    viscosity_value        : 0.001
    the unit                :-
permeability   :
    permeability_value     : 0.0000
    the unit                :-
the thermal_properties :
    the thermo_diffusion_coeff :
        thermo_diffusion_coeff_value : 0.0000
        the unit                       :-
    the thermal_conductivity :
        thermal_conductivity_value     : 0.0000
        the unit                       :-
    the heat_transfer_coeff :
        heat_transfer_coeff_value      : 0.0000
        the unit                       :-
    the specific_heat_capacity :
        specific_heat_capacity_value    : 0.0000
        the unit                       :-
    the emissivity      :
        emissivity_value                : 0.0000
        the unit                       :-
    the volume_expansion_coeff :
        volume_expansion_coeff_value    : 0.0000
        the unit                       :-

the mechanical_properties :
    the slip_coeff :
        the slip_coeff_value            : 0.0000
        the unit                       :-

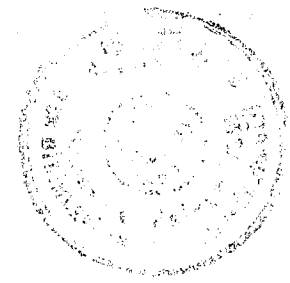
CFD DOMAIN GEOMETRY :
solid_model :
    the csg_model                :-
    the solid_b_rep              :-

surface_model :
    the swept_surface            :-
    the elementary_surface       :-
    the bounded_surface         :-

curve      :
    the bounded_curve           :-
    the conic_curve             :-
    the line                     :-

vertex    :

```



2D\_problem :

key\_points : 1  
topological\_coordinates :  
  the i\_index : 3  
  the j\_index : 1  
  the k\_index : 1

cartesian\_points :  
  the x\_coordinate : 1.0000  
  the y\_coordinate : 0.0000

key\_points : 2  
topological\_coordinates :  
  the i\_index : 5  
  the j\_index : 1  
  the k\_index : 1

cartesian\_points :  
  the x\_coordinate : 2.0000  
  the y\_coordinate : 0.0000

key\_points : 3  
topological\_coordinates :  
  the i\_index : 5  
  the j\_index : 3  
  the k\_index : 1

cartesian\_points :  
  the x\_coordinate : 1.7300  
  the y\_coordinate : 0.3300

key\_points : 4  
topological\_coordinates :  
  the i\_index : 5  
  the j\_index : 5  
  the k\_index : 1

cartesian\_points :  
  the x\_coordinate : 1.4500  
  the y\_coordinate : 0.6600

key\_points : 5  
topological\_coordinates :  
  the i\_index : 5  
  the j\_index : 7  
  the k\_index : 1

cartesian\_points :

```

the x_coordinate      : 1.0000
the y_coordinate      : 0.0000

key_points            : 1
topological_coordinates :
  the i_index         : 3
  the j_index         : 1
  the k_index         : 1

cartesian_points      :
  the x_coordinate     : 1.1700
  the y_coordinate     : 1.0000

key_points            : 6
topological_coordinates :
  the i_index         : 3
  the j_index         : 7
  the k_index         : 1

cartesian_points      :
  the x_coordinate     : 1.0000
  the y_coordinate     : 1.0000

key_points            : 7
topological_coordinates :
  the i_index         : 3
  the j_index         : 5
  the k_index         : 1

cartesian_points      :
  the x_coordinate     : 1.0000
  the y_coordinate     : 0.6600

key_points            : 8
topological_coordinates :
  the i_index         : 3
  the j_index         : 3
  the k_index         : 1

cartesian_points      :
  the x_coordinate     : 1.0000
  the y_coordinate     : 0.3300

key_points            : 9
topological_coordinates :
  the i_index         : 3
  the j_index         : 5
  the k_index         : 1

```

```

cartesian_points      :
  the x_coordinate    : 1.0000
  the y_coordinate    : 0.6600

key_points            : 10
topological_coordinates :
  the i_index         : 1
  the j_index         : 5
  the k_index         : 1

cartesian_points      :
  the x_coordinate    : 0.0000
  the y_coordinate    : 0.6600

key_points            : 11
topological_coordinates :
  the i_index         : 1
  the j_index         : 3
  the k_index         : 1

cartesian_points      :
  the x_coordinate    : 0.0000
  the y_coordinate    : 0.3300

key_points            : 12
topological_coordinates :
  the i_index         : 3
  the j_index         : 3
  the k_index         : 1

cartesian_points      :
  the x_coordinate    : 1.0000
  the y_coordinate    : 0.3300

key_points            : 13
topological_coordinates :
  the i_index         : 1
  the j_index         : 1
  the k_index         : 1

cartesian_points      :
  the x_coordinate    : 0.0000
  the y_coordinate    : 0.0000

3D_problem :
  keypoints          : 0
  logical_space_points :

```

```

index_i           : 0
index_j           : 0
index_k           : 0

cartesian_coordinate_points :
  the coordinate_x : 0.0000
  the coordinate_y : 0.0000
  the coordinate_z : 0.0000

keypoints         : 0
logical_space_points :
  index_i         : 0
  index_j         : 0
  index_k         : 0

cartesian_coordinate_points :
  the coordinate_x : 0.0000
  the coordinate_y : 0.0000
  the coordinate_z : 0.0000

```

#### CFD BOUNDARY AND INITIAL CONDITIONS :

```

boundary_conditions :-
initial_conditions  :-

```

#### SOLUTION TECHNIQUES AND STRATEGIES :

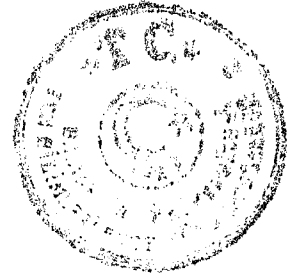
```

steady_state :
  linear :
    direct_gauss_elimination :-
    iterative_methods :
      preconditioning :
        relaxation_type :-
        diagonal_scaling :-
        incomplete_factorization :-

    symmetric_coefficient_matrix :
      conjugate_gradient :-
      conjugate_residual :-

    nonsymmetric_coefficient_matrix :
      conjugate_gradient_squared :-
      generalized_minimum_residual :-

```



```
non_linear :
  fully_coupled :
    solution_methods :
      successive_substitution :
        successive_substitution      :-
        iteration_number             :0

      newton_raphson_method :
        newton_raphson_method       :-
        iteration_number             :0

      quasi_newton_method :
        quasi_newton_method         :selected
        iteration_number             :10

      modified_newton_method :
        modified_newton_method       :-
        iteration_number             :0

  solution_strategies :
    successive_substitution :
      successive_substitution      :selected
      iteration_number            :7

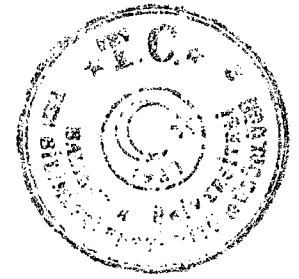
    newton_raphson_method :
      newton_raphson_method       :-
      iteration_number            :0

    quasi_newton :
      quasi_newton_method         :-
      iteration_number            :0

    modified_newton_method :
      modified_newton_method       :-
      iteration_number            :0

  segregated :
    boundary_surface_type_fixed :
      pressure_projection          :-
      pressure_correction          :-
      pressure_update              :-
    iterative_methods :
      preconditioning :
        relaxation_type           :-
        diagonal_scaling          :-
```





```
incomplete_factorization      :-

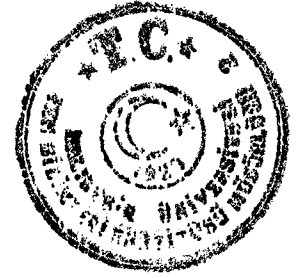
symmetric_linear_eq_sys :
  conjugate_gradient           :-
  conjugate_residual           :-

nonsymmetric_linear_eq_sys    :-

boundary_surface_type_free :
  kinematic_iteration         :-
  normal_stress_iteration     :-

transient :
  linear_problem :
    direct_gauss_elimination   :-
    iterative_methods :
      preconditioning          :
        relaxation_type        :-
        diagonal_scaling       :-
        incomplete_factorization :-
      symmetric_coeff_matrix   :
        conjugate_gradient     :-
        conjugate_residual     :-
    nonsymmetric_linear_eq_sys_ :
      conjugate_gradient_squared :-
      generalized_minimum_residual :-

non_linear_problem :
  implicit_time_integration :
    first_order_backward_euler_time_integration :-
    second_order_trapezoid_time_integration :-
  explicit_time_integration :
    forward_euler_scheme      :-
    subcycling_stratgy        :-
```

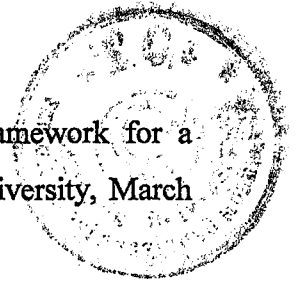


## KAYNAKÇA

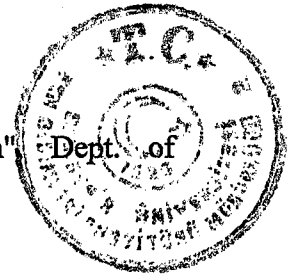
- [1] FRENCH, M.J., " Engineering Design : The Conceptual Stage", Heinemann Educational Books Ltd., London, 1971.
- [2] PAHL, G. and BEITZ, W. " Engineering Design : A Systematic Approach", The Design Council, London, 1988.
- [3] SUH, N.P., "The Principles of Design", Oxford University Press, New York, 1990.
- [4] SHEPHARD, M.S., YERRY, A.M., "Toward automated finite element modeling for the unification of engineering design and analysis", Finite Element in Analysis and Design 2 (1986) , pp. 143-160, Nort Holland.
- [5] TWORZYDLO, W.W., ODEN J.T., "Towards an automated environment in computational mechanics", Computer Methods in Applied Mechanics and Engineering, 104, 1993, pp. 87-143.
- [6] DYM, C.L., " Engineering Design : A Synthesis of Views" Cambridge University Press, 1994.
- [7] UGHANWA, D.O. "The Role of Design in International Competitiveness", Routledge, 1989, ISBN 0-415-00013-0.
- [8] DYM, C.L. and LEVITT, R.E. " Knowledge-Based Systems in Engineering", McGraw-Hill, New York, 1991.



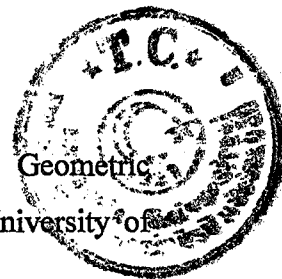
- [9] WOODSON, T.T., "Introduction to Engineering Design", McGraw-Hill, New York, 1966.
- [10] FINGER, S., DIXON, J.R. " A Rivew of Research in Mechanical Engineering Design. Part II : Representations, Analysis, and Design for the Life Cycle", Research in Engineering Design, Vol 1, pp.121-137, 1989.
- [11] HORNBY, A.S., "Oxford Advanced Learner's Dictionary", Oxford University Press, 1989, ISBN 0 19 431110 4.
- [12] BROOKE, D.M., " A Framework for Information in Engineering Analyses", PhD Thesis, The University of Leeds, May 1994.
- [13] SIMON, H.A., "A Student's Introduction to Engineering Design", Pergamon Press, 1975, ISBN 0-08-018234-8.
- [14] DIXON, J.R., " Design Engineering: Inventiveness, Analysis, and Decision Making", McGraw Hill Book Company, 1966.
- [15] BEAKLEY, C.G., CHILTON, E.G., "Design : Serving the needs of man", MacMillan Publishing Company, 1974, pp.285-286, ISBN 0-02-307240-7.
- [16] SHEPHARD, M.S., KORNGOLD E.V., WENTORF, R., "Design Systems Supporting Engineering Idealizations", Geometric Modeling for Product Engineering, North Holland, 1990, pp.279-300.
- [17] FINN, D., " Introduction : Preliminary Stages of Engineering Analysis and Modeling", in AI EDAM, 1993, 7 (4), pp. 231-237.
- [18] De PENNINGTON, A., and Bell, P. "Integrated Design and Manufacture", Seminar, Dept. of Manufacturing Engineering, Loughborough University of Technologies, March 1990.



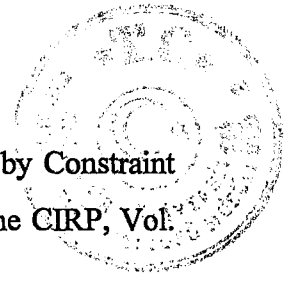
- [19] MCKAY, A., BLOOR, M.S., De PENNINGTON,A., "A Framework for a Product Data Model", ISS report series Number 19, Leeds University, March 1991.
- [20] MORTENSON, M.E., "Geometric Modelling", John Wiley \ & Sons, 1985, New York.
- [21] REQUICHA, A.A.G., "Representations for Rigid Solids:Theory,Methods and System", Computing Surveys, Vol:12, No.4, 1980.
- [22] REQUICHA, A.A.G., VOELCKER,H.B., "Solid Modeling: Current Status and Research Directions" , IEEE CG and A.,October 1983, pp. 25-37.
- [23] REQUICHA, A.A.G., "Solid Modelling and its applications : Progress in 1990", Proceedings of 17th NSF Design and Manufacturing Systems Grantees Conference, January 9-12, 1991, University of Texas, Austin.
- [24] ZEID, I., "CAD/CAM Theory and Practice", McGraw-Hill Inc., 1991and Systems" , Computing Surveys, Vol:12, No.4, December 1980.
- [25] REQUICHA, A.A.G., VOELCKER H.B., "Solid Modeling:A Historical Summary and Contemporary Assessment" ,IEEE CG and A, March 1982, pp. 9-26
- [26] REQUICHA, A.A.G., VOELCKER,H.B., "Boolean Operations in Solid Modelling : Boundary evaluation and merging algorithms", Proceedings of the IEEE, vol. 73, pp. 30-44, January 1985.
- [27] MÄNTYLÄ, M., "An Introduction to Solid Modelling", Computer Science Press, 1993.



- [28] Geometric Modelling Project "NONAME Documentation" Dept. of Mechanical Eng. University of Leeds, September 1983.
- [29] PRATT, M.J. "Representation and Communication in Geometric Modelling", Cranfield Institute of Technology, Cranfield, U.K. pp. 525-544.
- [30] SHAW, N.K., BLOOR, M.S., De PENNINGTON, A. "Product Data Models", Research in Engineering Design, Springer-Verlag, New York, 1989, pp.43-50.
- [31] BLOOR, M.S., De PENNINGTON, A., HARRIS, S.B., HOLDSWORTH, D. MCKAY, A., SHAW, N.K. "Towards Integrated Design and Manufacturing", International Conference on Factory 2000, Integrating Information and Material Flow, Churchill College, Cambridge, 31 August-2 Sept. 1988, pp 21-29.
- [32] MCKAY, A. "A Framework for the Project Meta-structure", Department of Mechanical Engineering, University of Leeds and Department of Manufacturing Eng. Loughborough University of Technology, 1989, iss-pds-report-8.
- [33] ISO EXPRESS Comitee, EXPRESS Language Reference Manual, ISO TC184/SC4/WG5, Document N9.
- [34] MCKAY, A. "Levels of Instance in Product Models", Department of Mechanical Engineering, University of Leeds and Department of Manufacturing Eng. Loughborough University of Technology, Report Series Number 25, January 1991.
- [35] MÄNTYLÄ, M., "Directions For Research in Product Modeling", Computer Applications in Production and Engineering, F.Kimura and A. Rolstadas(Eds), Elsevier Science Publishers , North Holland, pp. 71-85, 1989



- [36] MCKAY, A., HOLDSWORTH, D. "The Structure Editor", Geometric Modelling Project, user-12, Dept. of Mechanical Engineering, University of Leeds, 1987.
- [37] HOLDSWORTH, D. " $\lambda$ -Calculus Capability in a Structure Editor", iss-report-series-1, Sept. 1988, Dept. of Mechanical Engineering, University of Leeds.
- [38] MCKAY, A., HOLDSWORTH, D. "The Structure Editor", Geometric Modelling Project, user-12, Dept. of Mechanical Engineering, University of Leeds, 1987.
- [39] MCKAY, A. " User Manual For : Technical for Using the Structure Editor", Technical Report, University of Leeds and Loughborough University of Technology, February 1990, ISS Project, iss-user-2.
- [40] MCKAY, A. " User Manual For : Update to the GMP2 Structure Editor User Manual (USER12)", University of Leeds and Loughborough University of Technology, July 1989, iss-user-1.
- [41] FARAJ, I.Z., "Code generation and Driving an Application", cae-se-course-3, Department of Mechanical Engineering, University of Leeds, October 1993.
- [42] WICKENS, L.P., " A Syntax for Dimensions and Tolerances", PhD thesis, Department of Mechanical Engineering, University of Leeds, August 1990.
- [43] YOUNG, R.I.M., "Machine Planning in a Product Model Environment", PhD thesis, Loughborough University of Technology, 1991.
- [44] SATA,T., KIMURA,F., SUZUKI, H., AND FUJITA, T., "Designing Machine Assembly Structure Using Geometric Constraints in Product Modelling", Annals of the CIRP Vol34, (1), 1985, 169-172.

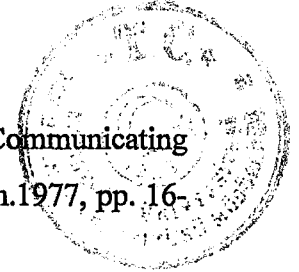


- [45] KIMURA, F., AND SUZUKI, H., "Variational Product Design by Constraint Propagation and Satisfaction in Product Modelling", *Annals of the CIRP*, Vol. 35 (1), 75-78, 1986.
- [46] SHAH, J.J., ROGERS, M.T., SREEVALSAN, P.C., HSIAO, D.W., MATHEW, A., BHATNAGAR, A., LION, B.B., AND MILLER, D.W., "The A.S.U. Feature Testbed : An Overview", *Computer in Engineering*, Vol. (1), ASME pp.233-241, 1990.
- [47] SPUR, G., KRAUSE, F.L. AND ARMBRUST, P. "Product models as basis for integrated design and manufacturing", *Int. Journal of Machine Tool Design and Research*, 26(2), pp.171-178, 1986
- [48] GRABOWSKI, H., ANDERL, R., AND PRATT, M.J., *Advanced Modelling for CAD/CAM Systems*", Springer-Verlag, Research Report, ESPRIT Project 322 , CAD Interfaces CAD\*I Vol 7, 1991.
- [49] GU, P., AND CHAN, K., "Product Modelling Using STEP", *Computer Aided Design*, Vol. 27, No:3, pp. 163-179, March 1995.
- [50] KIMURA, F., AND SUZUKI, H., "A Uniform approach to Dimensioning and Tolerancing in Product Modelling", *Minutes Process Planning Program at Gaitherburg, Arlington Texas, March 1986, Computer Aided Manufacturing-International Inc.*
- [51] KIMURA, F., AND SUZUKI, H., ANDO, H., SATO, H., KINOSADA A., "Variational Geometry BASED on Logical Constraints and its Applications to Product Modelling", *Annals of the CIRP* 36(1), pp.65-68, 1987.
- [52] OWEN, J. "STEP : An Introduction", *Information Geometers Ltd. 1993, ISBN 1-874728-04-6*

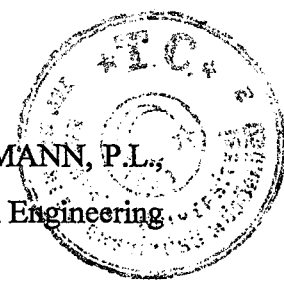


- [53] WILSON, P. R.,WOZNY,M.J., PRATT,M.J. (Eds), "A View of STEP" in Geometric Modeling for Product Realization, Elsevier Science Publishers, 1993 IFIP.
  
- [54] GOULT, R.J., "Towards a New Standard for Product Data Exchange", Proceedings 4th CIM Europe Conf. 18-20 May 1988, pp. 51-58, ISBN 0-948507-99-3.
  
- [55] THOMAS, D.,VAN MAANEN J., MEAD MICHAEL(Eds), ESPRIT Project 322 : CAD\*I Specification for Exchange of Product Analysis Data, Version 2, WG6.RAL.88.001, February 1988.
  
- [56] TRAPP, G., The Emerging STEP Standard for Product Model Data Exchange, Computer, 1993, pp. 85-87
  
- [57] ISO STEP DP 10303 " Standard for the Exchange of Product Model Data", October 1988, ISO TC184/SC4/WG1.
  
- [58] OWEN, J., BLOOR, S., "Neutral Formats for Product Data Exchange : The current situation.", Computer Aided Design, 19 (8), pp.436-443, October 1987.
  
- [59] Design/IDEF User's Manual, Version 2.5 Enhancements for the XWindows System, Meta Software Corp. 1993
  
- [60] ROSS, D.T., "Reflections on Requirements", IEEE Trans. Software Engineering, Vol. SE-3, No.1, Jan.1977, pp. 2-5.
  
- [61] ROSS, D.T., SCHOMAN, K.E., "Structured Analysis for Requirements Definition", IEEE Trans. Software Engineering, Vol. SE-3, No.1, Jan.1977, pp. 6-15





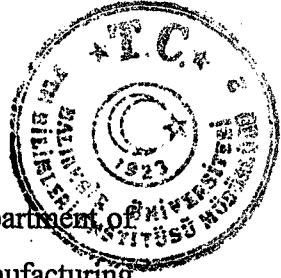
- [62] ROSS, D.T., "Structured Analysis (SA): A Language for Communicating Ideas", IEEE Trans. Software Engineering, Vol. SE-3, No.1, Jan.1977, pp. 16-34.
- [63] ROSS, D.T., "Applications and Extensions of SADT", IEEE Computer, April 1985, pp. 25-34.
- [64] ROSS, D.T., BRACKETT, J.W., BRAVOCO,R.R., SCHOMAN Jr. K.E., "Architect's Manual : ICAM Definition Method IDEF0", September 1979, SofTech Inc. Massachusetts.
- [65] KRAUSE, F.L., JANSEN, H., "Advanced Geometric Modelling for Engineering Applications", Proceedings of the IFIP WG 5.2/GI, International Symposium on Advanced Geometric Modelling for Engineering Applications, West Berlin, 8-10 Nov. 1989, North Holland, 1990, pp. 408, ISBN 0-444-88830-6.
- [66] HARRINGTON, J.Jr., "Understanding the Manufacturing Process : Key to Successful CAD/CAM implementation", Marcel Dekker Inc., 1984, New York, ISBN 0-8247-7170-2.
- [67] MARCA,D.A., MCGOWAN, CLEMENT L., "Structured Analysis and Design Technique", McGraw-Hill Book Company, 1987.
- [68] SHEPHARD, M.S., BAEHMANN, P.L., GEORGES M.K., KORNGOLD, E.V. "Framework for the Reliable Generation and Control of Analysis Idealizations", Computer Methods in Applied Mechanics and Engineering, Vol. 82, pp.257-280, 1990.
- [69] SHEPHARD, M.S. "Idealization in Engineering Modeling and Design, Research in Engineering Design, Vol. 229-238, 1990.



- [70] SHEPHARD, M.S., KORNGOLD E.V., COLLAR, R.R., BAEHMANN, P.L.,  
"A Modeling Framework for Controlling Structural Idealizations in Engineering  
Design, Computers & Structures, Vol. 37, pp. 181-191, 1990.
- [71] KORNGOLD, E.V., SHEPHARD, M.S., WENTORF, R., SPOONER, D.L.,  
"Architecture of a Design System for Engineering Idealizations", Computer  
Aided and Computational Design ASME NY, Advances in Design Automation  
1989 Vol. 1) pp. 259-265.
- [72] SZABÓ, B. A., "Geometric Idealizations in Finite Element Computations" ,  
Communications in applied numerical methods, Vol. 4, pp. 393-400, 1988.
- [73] SHEPHARD, M.S., WENTORF, R., "Toward the Implementation of  
Automated Analysis Idealization Control", Applied Numerical Mathematics,  
Vol. 14, pp. 105-124, 1994.
- [74] SZABÓ, B. A., BABUŠKA, I., "Finite Element Analysis", John Wiley & Sons  
Inc., 1991, ISBN 0-471-50273-1.
- [75] SCHENK, D., WILSON, P., "Information Modeling : The Express Way"  
Oxford Uni. Press, 1994.
- [76] VERYARD, R., "Information Modelling : Practical Guidance", Prentice Hall  
International (UK) Ltd., 1992, ISBN 0-13-454182-0.
- [77] ARABSHAHI, S., "Finite Element Idealisation in a Solid Modeling  
Environment", University of Leeds, Mech. Eng.Dept, PhD Thesis, 1991.
- [78] CROSS, N., "Engineering Design Methods", John Wiley & Sons, 1991, ISBN  
0-471-92215-3.



- [79] VENNARD, J.K., STREET, R.L., "Elementary Fluid Mechanics" 6th edition, John Wiley & Sons, 1982.
- [80] FIDAP 7.0 Fluid Dynamics Analysis Package Theory Manual", Fluid Dynamics International Inc., April 1993.
- [81] WILSON, P. R., "Euler formulas and geometric modeling", IEEE CG & A., pp. 24-36, August 1985.
- [82] DYM, C.L., LEVITT, R.E., "Toward the Integration of Knowledge for Engineering Modeling and Computation", Engineering with Computers, 7, pp. 209-224, 1991.
- [83] DAWSON, P., "The Horses User Interface to the Product Data Editor", Technical Report, The University of Leeds and Loughborough University of Technology, July 1990, ISS Project user manual, iss-user-4.
- [84] MCKAY, A., "The Structure Editor Approach to Product Description", Department of Mechanical Engineering, University of Leeds, 1988, iss-pds-report-4.
- [85] ICHIBAH, J.D., et al "Reference Manual for the ADA Programming Language, ANSI/MIL-STD 1815 A-1983, United State Department of Defense, Castle House Publication Ltd. Kent, U.K. Reprinted 1987.
- [86] CLARK, R.G., "Programming in ADA : a first course", Department of Computer Science, University of Stirling, Cambridge University Press, 1985.
- [87] FARAJ, I.Z., MCKAY A., "How to Use the Structure Editor's Automatic Code Generator", Department of Mechanical Engineering, University of Leeds, 1992, cae-user-1.



- [88] MCKAY, A., "The Generation of Code for Building Instances", Department of Mechanical Engineering, University of Leeds and Department of Manufacturing Eng. Loughborough University of Technology, 1989, iss-software-8-fd-1.
- [89] ISO Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part 11 : The EXPRESS Language Reference Manual, ISO DIS 10303-11, ISO TC184/SC4 N151, 1992.
- [90] GÜLESİN, M., "The STEP Standard and Data Exchange Between CAD/CAM Systems", MATİK'97 Makina Tasarım Teorisi ve Modern İmalat Yöntemleri Konferansı, G.Ü. Teknik Eğitim Fakültesi, 15-16 Eylül 1997, pp.281-294, ANKARA.
- [91] DERELİ, T., FİLİZ, İ.H., "Bilgisayar Destekli Tasarım ve Üretim Sistemlerinde Grafik Standartlar", MATİK'97 Makina Tasarım Teorisi ve Modern İmalat Yöntemleri Konferansı, G.Ü. Teknik Eğitim Fakültesi, 15-16 Eylül 1997, pp. 258-265, ANKARA.