

**TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**ÜÇ BOYUTLU, YAPISAL OLMAYAN ÇÖZÜM AĞLARI ÜZERİNDE, SIKIŞTIRILABİLİR  
REAKTİF AKIŞLAR İÇİN GPU DESTEKLİ, PARALEL, HESAPLAMALI AKIŞKANLAR  
DİNAMİĞİ ÇÖZÜCÜSÜ GELİŞTİRİLMESİ**

**YÜKSEK LİSANS TEZİ**

**Bertan ÖZKAN**

**Makine Mühendisliği Anabilim Dalı**

**Tez Danışmanı: Dr. Öğr. Üyesi Sıtkı USLU**

**TEMMUZ 2018**



Fen Bilimleri Enstitüsü Onayı

.....  
**Prof.Dr. Osman EROĞUL**  
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığımı onaylarım.

.....  
**Doç. Dr. Murat Kadri AKTAŞ**  
Anabilimdalı Başkanı

TOBB ETÜ, Fen Bilimleri Enstitüsü'nün 141511043 numaralı Yüksek Lisans öğrencisi **Bertan ÖZKAN** 'nın ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "**ÜÇ BOYUTLU, YAPISAL OLMAYAN ÇÖZÜM AĞLARI ÜZERİNDE, SIKIŞTIRILABİLİR REAKTİF AKIŞLAR İÇİN GPU DESTEKLİ, PARALEL, HESAPLAMALI AKIŞKANLAR DİNAMİĞİ ÇÖZÜCÜSÜ GELİŞTİRİLMESİ**" başlıklı tezi **02.07.2018** tarihinde aşağıda imzaları olan jüri tarafından kabul edilmiştir.

**Tez Danışmanı:** **Dr. Öğr. Üyesi Sıtkı USLU** .....  
TOBB Ekonomi ve Teknoloji Üniversitesi

**Jüri Üyeleri:** **Prof.Dr. Yusuf ÖZYÖRÜK (Başkan)** .....  
Orta Doğu Teknik Üniversitesi

**Doç.Dr. Murat Kadri AKTAŞ** .....  
TOBB Ekonomi ve Teknoloji Üniversitesi

**Doç. Dr. Özgür Uğraş BARAN** .....  
Orta Doğu Teknik Üniversitesi

**Prof.Dr. İsmail Hakkı TUNCER** .....  
Orta Doğu Teknik Üniversitesi



## TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

Bertan ÖZKAN



## ÖZET

Yüksek Lisans Tezi

ÜÇ BOYUTLU, YAPISAL OLMAYAN ÇÖZÜM AĞLARI ÜZERİNDE,  
SIKIŞTIRILABİLİR REAKTİF AKIŞLAR İÇİN GPU DESTEKLİ, PARALEL,  
HESAPLAMALI AKIŞKANLAR DİNAMİĞİ ÇÖZÜCÜSÜ GELİŞTİRİLMESİ

Bertan ÖZKAN

TOBB Ekonomi ve Teknoloji Üniversitesi

Fen Bilimleri Enstitüsü

Makine Mühendisliği Anabilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Sıtkı USLU

Tarih: TEMMUZ 2018

Bu tez kapsamında üç boyutlu, yapısal olmayan çözüm ağları üzerinde çalışan, sıkıştırılabilir, kimyasal reaksiyonlu ve birden çok kimyasal türe sahip akış problemlerini; GPU donanımı desteği ile CPU donanımında paralel olarak çözebilen bir hesaplamalı akışkanlar dinamiği çözücüsü geliştirilmiştir.

Söz konusu çözücü basınç temelli çözüm algoritmasını kullanmaktadır. Eş konumlu yapısal olmayan sayısal ağlarında çözüm yapabilmektedir. Çözücü, zamana bağlı veya zamandan bağımsız, Euler ve Navier-Stokes denklemlerini çözebilmektedir. Çözücü, sonlu hacimler yöntemi kullanıyor olup birinci mertebeden konveksiyon ayrıştırmasına sahiptir. Çözücü, AMG (Algebraic Multigrid) yöntemleri ile desteklenen iteratif seyrek matris çözücüleri sayesinde zamanda kapalı (implicit) olarak çözüm yapabilmektedir. Çözücü, Arrhenius tipi kimyasal reaksiyon modeli ve birden fazla kimyasal tür için çözüm sunmaktadır. Çözücünün içerisinde ideal gaz denklemi kullanılmaktadır. Özgül ısı polinomlar şeklinde sıcaklığın fonksiyonu olarak hesaplanmaktadır. Çözücü içerisinde kullanılan üçüncü parti matris çözüme yazılımı ViennaCL sayesinde çözücü GPU donanımında paralel olarak çalışabilmektedir.

Çözücü, sıkıştırılmaz akış problemlerinin yanında sıkıştırılabilir akışlar için de kullanılabilir. Doğrulama çalışmaları dahilinde sıkıştırılmaz akış için düz plaka üzerinde laminar akış problemi, sıkıştırılabilir akış için de tümsek üzerinde ses altı ve ses üstü akış problemleri çözülmüştür. Zamana bağlı problemlerin doğrulaması için de şok tüp problemi çözülmüştür. Reaktif problemler için de Sandia laboratuvarlarında yapılmış olan Flame D problemi çözülmüştür.

Çözücünün paralel hesaplama performansı ölçülmüştür. Çözücü GPU performansı ölçümünde, aynı fiyata sahip CPU donanımı ile GPU donanımı aynı problem ile denenmiştir. Çözücü GPU modunda iken bu problemin matrislerini, veri transferi süreleri dahil aynı fiyattaki CPU donanımından üç buçuk kat daha hızlı çözdüğü görülmüştür.

**Anahtar Kelimeler:** Reaktif hesaplamalı akışkanlar dinamiği, GPU destekli hesaplama.



## ABSTRACT

Master of Science

### DEVELOPMENT OF A THREE DIMENSIONAL, UNSTRUCTURED, GPU ACCELERATED, PARALLEL COMPUTATIONAL FLUID DYNAMICS SOLVER FOR COMPRESSIBLE REACTING FLOWS

Bertan ÖZKAN

TOBB University of Economics and Technology  
Institute of Natural and Applied Sciences  
Department of Mechanical Engineering

Supervisor: Dr. Öğr. Üyesi Sıtkı USLU

Date: JULY 2018

In this thesis a three dimensional, compressible solver is developed. It solves three dimensional compressible chemically reacting flows with multiple species. The solver can operate on parallel CPU's with the help of GPU hardware.

It is a pressure based methodology that can operate on unstructured co-located grids. It solves transient Euler and Navier-Stokes Equations. Solver uses Finite Volume Method and it uses a first order upwind discretization for convective terms. With the help of iterative sparse matrix solvers supported by AMG smoother, it can solve problems implicitly. Arrhenius type chemical reaction rates are used with multiple species. Solver uses ideal gas equation as equation of state. Variables like specific heat is a function of temperature in the solver. With the third party matrix solver ViennaCL this solver can perform parallel computations on GPU hardware.

In an effort to validate the solver, laminar flow over flat plate problem, subsonic, transonic and supersonic flow over circular arc bump problem, Sod's shock tube problem and Flame D test case from Sandia laboratories are studied.

Computing performance of the solver is also studied. In the GPU performance study, it is concluded that the solver can solve same matrix three and half times faster in similarly priced GPU than CPU, including data transfer times between main memory and GPU memory.

**Keywords:** Reactive computational fluid dynamics, GPU accelerated computing.



## TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren, beni CFD ile tanıştıran hocam Dr. Öğr. Üyesi Sıtkı USLU'ya, kıymetli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Makine Mühendislięi ve Bilgisayar Mühendislięi Bölümü öğretim üyelerine , yüksek lisans eğitimim boyunca bana her zaman destek olan çalışma arkadaşlarım Ozan Can KOCAMAN, Tekin AKSU, Yücel SAYĞIN, Serhan DÖNMEZ, Serkan Berkay KÖRPE, Mahmut DOĞRUDİL, Tacettin Utku SÜER, Burak CENİK ve dięer bütün CSL (Combustion Systems Laboratory) üyelerine, yüksek lisans eğitimim boyunca bana burs imkanı sağlayarak çalışmalarına destek olan TOBB Ekonomi ve Teknoloji Üniversitesine, ViennaCL, OpenMP, Linux, Latex, CUDA, GCC, Paraview ve NetBeans başta olmak üzere büyük uğraşlar sonucu hayata geçirdikleri projelerini karşılıksız olarak akademisyenlerin kullanımına açan bütün geliőtiricilere, destekleriyle her zaman yanımda olan, beni bilime yönlendiren aileme ve arkadaşlarıma çok teşekkür ederim.



## İÇİNDEKİLER

	<u>Sayfa</u>
<b>ÖZET</b> . . . . .	<b>iv</b>
<b>ABSTRACT</b> . . . . .	<b>vi</b>
<b>TEŞEKKÜR</b> . . . . .	<b>viii</b>
<b>İÇİNDEKİLER</b> . . . . .	<b>ix</b>
<b>ŞEKİL LİSTESİ</b> . . . . .	<b>xi</b>
<b>ÇİZELGE LİSTESİ</b> . . . . .	<b>xiii</b>
<b>KISALTMALAR</b> . . . . .	<b>xiv</b>
<b>SEMBO LİSTESİ</b> . . . . .	<b>xv</b>
<b>1. GİRİŞ</b> . . . . .	<b>1</b>
1.1 Tezin Amacı . . . . .	1
1.2 Literatür Araştırması . . . . .	1
<b>2. ÇÖZÜCÜDE KULLANILAN YÖNTEMLER</b> . . . . .	<b>7</b>
2.1 Fiziksel Problemin Çözümünde Kullanılan Denklemler . . . . .	7
2.2 Sonlu Hacimler Yöntemi . . . . .	10
2.3 Basınç Merkezli Çözüm Algoritması . . . . .	15
2.4 Türbülansın Modellenmesi . . . . .	22
2.5 Kimyasal Reaksiyonların Modellenmesi . . . . .	26
2.6 Sınır Koşullarının Uygulanması . . . . .	28
2.7 Doğrusal Denklem Sistemi Çözümü için Kullanılan Metodlar . . . . .	30
2.7.1 Successive over-relaxation Matris Çözme Metodu . . . . .	30
2.7.2 Algebraic Multigrid metodu . . . . .	31
2.7.3 Conjugate Gradient Matris çözme metodu . . . . .	32
2.8 Kullanılan Ek Metotlar . . . . .	33
2.8.1 Gauss Green Gradyan Yapılandırma Metodu . . . . .	33
2.8.2 Rhie and Chow Yüzey İnterpolasyonu Metodu . . . . .	33

<b>3. ÇÖZÜCÜNÜN GELİŞTİRİLMESİNDE KULLANILAN YÖN- TEMLER</b> . . . . .	35
3.1 Nesne Tabanlı Programlama . . . . .	35
3.2 Girdi-Çıktı Dosyaları . . . . .	37
3.3 Paralel Programlama Metodları . . . . .	38
<b>4. DOĞRULAMA ÇALIŞMALARI</b> . . . . .	41
4.1 İki boyutlu tümsek üzerinde akış . . . . .	41
4.1.1 İki boyutlu tümsek üzerinde ses altı akış . . . . .	41
4.1.2 İki boyutlu tümsek üzerinde transonik akış . . . . .	42
4.1.3 İki boyutlu tümsek üzerinde ses üstü akış . . . . .	44
4.2 Düz plaka üzerinde laminar akış . . . . .	46
4.3 Sod Shock Tube doğrulama çalışması . . . . .	50
4.4 Sandia Flame D doğrulama çalışması . . . . .	53
<b>5. PERFORMANS ÖLÇÜMÜ ÇALIŞMALARI</b> . . . . .	61
<b>6. SONUÇ VE ÖNERİLER</b> . . . . .	63
<b>ÖZGEÇMİŞ</b> . . . . .	69

## ŞEKİL LİSTESİ

	<u>Sayfa</u>
Şekil 2.1: Komşu olan iki sonlu hacimler hücresinin değişkenleri . . . . .	12
Şekil 3.1: Çözücüdeki nesne sınıfları . . . . .	36
Şekil 4.1: Tümsek üzerinde ses altı akış Mach sayısı eş eğrileri. . . . .	42
Şekil 4.2: Tümsek üzerinde ses altı akış alt duvar Mach sayısı karşılaştırılması. . . . .	42
Şekil 4.3: Tümsek üzerinde ses altı ve transonik akış için çözüm ağı. . . . .	43
Şekil 4.4: Tümsek üzerinde transonik akış Mach sayısı eş eğrileri. . . . .	43
Şekil 4.5: Tümsek üzerinde transonik akış alt duvar Mach sayısı karşılaştırılması. . . . .	44
Şekil 4.6: Tümsek üzerinde ses üstü akış çözüm ağı. . . . .	45
Şekil 4.7: Tümsek üzerinde ses üstü akış Mach sayısı eş eğrileri. . . . .	45
Şekil 4.8: Tümsek üzerinde ses üstü akış alt duvar Mach sayısı karşılaştırılması. . . . .	46
Şekil 4.9: Düz plaka üzerinde laminar akış Mach sayısı eş eğrileri ölçekli şekli. . . . .	48
Şekil 4.10: Düz plaka üzerinde laminar akış çözüm ağı. . . . .	48
Şekil 4.11: Düz plaka üzerinde laminar akış sürtünme katsayısı (a) ve hız profili (b) karşılaştırması. . . . .	49
Şekil 4.12: Sod Shock tube problemi Basınç eş eğrileri. . . . .	51
Şekil 4.13: Sod shock tube problemi basınç karşılaştırması. . . . .	51
Şekil 4.14: Sod shock tube problemi MACH sayısı (a) ve yoğunluk (b) karşılaştı- rması. . . . .	52
Şekil 4.15: Flame D test düzeneği fotoğrafı [22] . . . . .	53
Şekil 4.16: Flame D test düzeneği şematiği [37] . . . . .	54
Şekil 4.17: Flame D problemi çözüm ağı ayrıntısı. . . . .	54
Şekil 4.18: Flame D problemi sıcaklık eş eğrileri. . . . .	56
Şekil 4.19: Flame D problemi $CO$ (a) ve $H_2O$ (b) Kütlelesel derişim eş eğrileri. . . . .	57

Şekil 4.20: Sandia Flame D $x = 7.2$ mm (a) , $x = 14.4$ mm (b) konumlarında sıcaklık karşılaştırılması. . . . .	58
Şekil 4.21: Sandia Flame D $x = 21.6$ mm (a) , $x = 108$ mm (b) konumlarında sıcaklık karşılaştırılması. . . . .	59
Şekil 5.1: CPU paralel durumda matris çözümü için harcanan zaman hariç hız artışı değerleri. . . . .	61
Şekil 5.2: CPU ve GPU donanımında matris çözüm süreleri [saniye]. . . . .	62





## ÇİZELGE LİSTESİ

	<u>Sayfa</u>
Çizelge 2.1: $k - \varepsilon$ türbülans modeli sabitleri . . . . .	25
Çizelge 4.1: Sod Shock Tube probleminin başlangıç koşulları. . . . .	50
Çizelge 4.2: Metan hava yanması global reaksiyon mekanizması [23]. . . . .	55





## KISALTMALAR

- HAD** : Hesaplamalı Akışkanlar Dinamiği  
**CFL** : Courant-Friedrichs-Lewy kriteri  
**CPU** : Merkezi İşlem Birimi (Central Processing Unit)  
**GPU** : Grafik İşlemci Ünitesi (Graphics Processing Unit)  
**RANS** : (Reynolds Averaged Navier-Stokes)



## SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

### Simgeler Açıklama

$c$	Ses Hızı
$c_p$	Sabit basınçta özgül ısı
$c_v$	Sabit hacimde özgül ısı
$\mathbf{u}$	Hız vektörü
$u_x$	X yönündeki hız
$u_y$	Y yönündeki hız
$u_z$	Z yönündeki hız
$\phi$	Taşıma denkleminde taşınan skaler değer
$\bar{\phi}$	Taşınan değişkenin zaman ortalamalı hali
$\Gamma$	Taşıma denkleminde difüzyon teriminin katsayısı
$T$	Sıcaklık
$\mu$	Viskozite
$\mu_t$	Türbülanslı viskozite
$S_{ij}$	Gerinim Tensörü
$P$	Basınç
$\rho$	Yoğunluk
$t$	Zaman
$\phi_P$	Çözüm yapılan hücrenin taşınan değişkeni
$\phi_A$	Çözüm yapılan hücrenin komşu hücresinin taşınan değişkeni
$\Delta A$	Hücre yüzeyinin alanı

$\Delta V$	Hücrenin hacmi
$\Delta \zeta$	Hücre merkezinin komşu hücre merkezine uzaklığı
$\mathbf{n}$	Hücre yüzeyinin normal vektörü
$\Delta t$	Zaman adımı
$\sigma_h$	Prandtl sayısı



# 1. GİRİŞ

## 1.1 Tezin Amacı

Bu tez kapsamında bütün Mach sayılarındaki sıkıştırılabilir kimyasal reaksiyonlu akış problemlerinin çözümü için; basınç merkezli, genel çözüm ağları üzerinde çalışan bir HAD çözücüsü geliştirilmiştir. Bu çözücü dahilinde kullanılan HAD ve sayısal yöntemlerden bahsedilmiştir. Bu çözücünün geliştirilmesinde kullanılan yüksek başarılı hesaplama yöntemleri gösterilmiştir. Doğrulama ve performans değerlendirilmesi çalışmaları yapılmıştır.

## 1.2 Literatür Araştırması

Patankar [6] makalesinde HAD analizlerinde kullanılmak üzere yoğunluk-basınç ve hız birleşimini sağlayan bir algoritma geliştirmiştir. Bu algoritma üç boyutlu parabolik akışlarda ısı transferi, kütle transferi ve momentum transferi çözümü için kullanılmaktadır. Bu makaledeki model çapraz konumlu çözüm ağlarında kullanılmak üzere geliştirilmiştir. Bu algoritmada momentum denkleminin çözümünden sonra basınç düzeltme adı verilen bir denklem çözülmektedir. Basınç düzeltme denklemden çıkan sonuçlar ile hız değişkenleri ile basınç değerleri düzeltilmektedir. Bu düzeltmeler yapıldıktan sonra yeni hesaplanan hız ve basınç parametreleri ile geri kalan denklemler çözülmektedir. Bu makale dahilindeki parabolik akış teriminin anlamı sayısal olarak tek bir yönde bilgi akışı olan akışlar olarak belirtilmiştir. Patankar bu algoritmayı doğrulamak için kesit alanının değiştiği bir kanala açılan dikdörtgen şeklinde bir girişin bulunduğu problemi ve kavite içindeki akış problemini çözmüştür. Patankar'ın bu çalışması sonucunda basınç temelli HAD algoritmalarının temeli atılmıştır ve SIMPLE adı verilen algoritma ortaya çıkmıştır. Patankar tarafından geliştirilen bu modeli diğer

modellerden ayıran özellik, bu modelin çok düşük hızlardaki sıkıştırılmaz akışları da çözebiliyor olmasıdır.

Issa [7] makalesinde Patankar'ın modeline benzeyen bir model geliştirmiştir. Issa'nın modelinin Patankar'ın modelinden en büyük farkı bir adet fazla düzeltme adımına sahip olmasıdır. Issa'nın modeline literatürde PISO adı verilmektedir. Bu model zamana bağlı olmayan sıkıştırılabilir akışları çözebilmektedir.

Demirdzic [5] makalesinde bütün MACH sayılarındaki sıkıştırılabilir akışları çözmek için basınç temelli bir çözüm algoritması geliştirmiştir. Demirdzic'in modeli de Patankar'ın ve Issa'nın modeli gibi basınç düzeltme mantığı ile geliştirilmiştir. Demirdzic'in modelindeki, akışkanın termodinamik özelliklerinden yola çıkarak hesaplanan bir terim, bu modelin yüksek MACH sayılarındaki akışları çözebilmesine olanak sağlamaktadır. Bu model eş konumlu çözüm ağlarında çalışmaktadır bu sayede yapısal olmayan çözüm ağları kullanabilmektedir. Demirdzic, bu modelini doğrulamak için %10 tümseğe sahip kanal içerisinde ses altı ve transonik akış problemlerini çözmüştür. Demirdzic, modelinin ses üstü akışlarda doğruluğunu göstermek için %4 tümseğe sahip kanal içerisinde ses ütü akış problemini, %4 kalınlıkta iki adet tümseğe sahip kanal içerisinde ses ütü akış problemini, çift boğazlı lüle problemini ve yüksek genişlemeli roket lülesi problemini çözmüştür. Bu problemlerin çözümünde kullanılan sayısal şema ikinci mertebe konveksiyon ayrıştırmasına sahiptir ve modelde adaptif çözüm ağı iyileştirmesi kullanılmıştır.

Mc Bride [10], makalesinde NASA desteği ile çok geniş bir akışkan özellikleri kütüphanesi oluşturmuştur. Bu akışkan kütüphanesi 5. mertebeden polinomlar kullanarak akışkanların özgül ısı ve oluşum entalpisi gibi özelliklerini sıcaklığa bağlı olarak kullanıcılara sunmaktadır.

Sutherland [11], makalesinde akışkanların viskozite değerlerini deneysel bir yöntem ile sıcaklığın fonksiyonu haline getirmiştir.

Barlow [22] , raporunda, Sandia laboratuvarları dahilinde yapılan ön karışım pilot



alevli Flame D testlerinin geometrik özelliklerini, sınır koşullarını ve deney sonuçlarını paylaşmıştır.

Alonso ve ekibi [27] dünyaca ünlü SU2 kodunu geliştirmişlerdir. SU2 kodu, sıkıştırılabilir akışları çözme amacı ile geliştirilmiş yoğunluk temelli bir HAD kodudur. SU2 kodu ikinci mertebeden konveksiyon şemasına sahiptir. Bu kod zamana bağlı HAD problemlerini çözebilmektedir. Bu kod ayrıca zamana bağlı olmayan HAD problemlerini lokal zaman adımı yöntemini kullanarak çözebilmektedir. SU2 kodu birçok yoğunluk tabanlı konveksiyon ayrıştırma şeması ile çözüm yapabilmektedir. SU2 kodu yakınsamayı hızlandırma amaçlı multigrad şemasına sahiptir. SU2 kodunun içinde kullanıcı gradyen yapılandırma yöntemlerinden Gauss-Green veya Least Squares seçebilmektedir. SU2 kodunun kapalı (implicit) zaman ayrıştırılabilir çözümleri için iteratif matris çözümleri vardır. Bu matris çözümlerinden bazıları LU-SGS, GMRES ve BICGSTAB çözümleridir. SU2 kodu genel olarak sesten hızlı dış akış problemlerinde kullanılmaktadır. Bu tarz problemlerin çözümüne uygun araçlar içermektedir. SU2 kodu şok yakalama amaçlı adaptif çözüm ağı iyileştirme yöntemine sahiptir. SU2 kodu dağıtılmış hafızaya sahip donanımlarda çözüm yapmak amacı ile MPI [31] protokolünde paralel olarak C++ dilinde yazılmıştır. Sonsuz ölçeklenebilirliğe sahip olması için çözüm ağını ayırma işlemini paralel yapabilen, açık kaynak kodlu PARMETIS [32] isimli yazılımı kullanmaktadır. SU2 yazılımı, açık kaynak kodlu bir yazılımdır ve kaynak kodu internet üzerinden yayınlanmaktadır [30].

Alonso ve ekibi [29] SU2 koduna türbülans modelleri de eklemiştir. SU2 kodu dış akış amaçlı bir kod olduğu için kodun içerisindeki türbülans modelleri dış akışa uygun türbülans modelleridir. Bu türbülans modelleri sıfır denklemlili Spalart-Allmaras türbülans modeli ve iki denklemlili SST  $k-\omega$  modelidir. Alonso ve ekibi SU2 kodunun doğrulaması için tümsek üzerindeki akış, NACA 12 airfoil, Onera Wing M6 ve DLR F6 gibi literatürde sıkça görülen doğrulama çalışmalarını yapmışlardır. SU2 yazılımının viskoz şemasının doğrulanması düz plaka üzerinde laminar akış problemi ile yapılmıştır.

Andrea Lani [33] doktora tezinde Coolfluid adlı, aero-termodinamik problemlerini çözüme amacı ile geliştirilmiş, yoğunluk temelli kimyasal reaksiyon çözüme kabiliyetli kodun geliştirilme aşamalarından ve yöntemlerinden bahsetmiştir. Bu kod yine SU2 gibi C++ dilinde nesne tabanlı olarak yazılmıştır. MPI protokolü ile paralelleştirilmiştir. SU2'den farklı olarak bu çözücünde kimyasal reaksiyon da çözülebilmektedir ve hal denkleminin içerisindeki özgül ısı değeri sıcaklığa bağlı olarak analize girilebilmektedir. Coolfluid çözücüsünün doğrulama çalışmaları Avrupa Uzay Ajansının Expert isimli atmosfere yeniden giriş aracının test verileri kullanılarak yapılmıştır. Bu çözücünün bir diğer doğrulama çalışması da yine bir atmosfere yeniden giriş aracı olan NASA Stardust aracının test verileri kullanılarak yapılmıştır. Bu çözücünün çok yüksek MACH sayılarındaki akışları çözebilme yeteneği içerisinde bulunan ısı ve kimyasal denge dışı çözücü modelinden gelmektedir. Bu çözücü denge dışı kimya çözmek için Arrhenius tipi kimya modeli kullanmaktadır. Termal denge dışı çözücüsü ise iki adet sıcaklık hesabı yapan TNE2 yöntemini kullanmaktadır.

Luke ve ekibi [34] genel çözüm ağları için kimyasal reaksiyon kabiliyetine sahip bir HAD çözücüsü geliştirmişlerdir. Bu çözücü kullanıcının girdiği herhangi bir sayıda reaksiyon ve kimyasal tür ile analiz yapabilmektedir. Bu çözücünde Spallart-Allmaras ve SST  $k-\omega$  türbülans modelleri vardır. Bu çözücü de yine diğer kimyasal reaksiyon kabiliyetine sahip çözücülerde olduğu gibi sıcaklığa bağlı özgül ısı değerleri ile analiz yapabilmektedir. Bu çözücünün doğrulama çalışmaları transonik Onera M6 kanadı ve Hidrojen yakıtlı deneysel RBCC yanma odası problemleri ile yapılmıştır.

EIGEN [35] , açık kaynak kodlu bir matris operasyonları kütüphanesidir. OpenMP ve MPI protokollerinde paralel olarak çalıştırılabilmektedir. HAD çözücülerinde sıklıkla kullanılan ConjugateGradient, BiCGSTAB ve Least Squares Conjugate Gradient gibi iteratif seyrek matris çözücülerini içerisinde barındırmaktadır.

ViennaCL [13], Viyana Teknik Üniversitesi tarafından geliştirilen, açık kaynak kodlu bir matris operasyonları kütüphanesidir. ViennaCL, CPU donanımında OpenMP protokolünde paralel olarak çalışabilmektedir. GPU donanımında ise CUDA veya OpenCL

kütüphanelerini kullanarak paralel çalışabilmektedir. ViennaCL ConjugateGradient ve BiCGSTAB gibi iteratif seyrek matris çözücülerinin yanında AMG [16] gibi yakınsama hızlandıran yöntemleri de kullanmaktadır. ViennaCL, akademik çalışmalarda sınırsız olarak kullanılabilir şekilde açık olarak lisanslandırılmaktadır. ViennaCL kaynak kodları ise internet üzerinden yayınlanmaktadır [36].

Poinsot [23], makalesinde, Büyük Burgaç Simülasyonu yöntemini kullanarak bir yanma odasındaki yanma kararsızlıklarını incelemiştir. Bu çalışmada global bir metan-hava reaksiyon mekanizması kullanılmıştır. Bu reaksiyon mekanizması 2 adet reaksiyona ve 6 adet kimyasal türe sahiptir.

MPI (Message Passing Interface) protokolü [31], dağıtılmış yüksek başarımli hesaplama sistemlerinde veya küme bilgisayar sistemlerinde paralel hesaplama yapabilmek amacı ile bilgisayarlar arası ağ sistemini kullanan bir protokoldür. MPI protokolü, hesaplama yapılan donanımlar arası mesaj iletimini kullanıcı dostu bir hale getirerek dağıtılmış hafızaya sahip sistemlerde paralel olarak çalışacak kod yazma işlemini kolaylaştırmıştır.

OpenMP protokolü [18], İşletim sistemi yazılımlarının çekirdeklerinde bulunan çoklu işlem yapma komutlarını tek bir protokol ve söz dizimi kuralları dahilinde toplamıştır. Bu sayede birçok işlemci çekirdeğinin paylaştığı hafıza kümesinin olduğu bilgisayar sistemlerinde CPU donanımında paralel kod geliştirme işlemini kolaylaştırmıştır. OpenMP protokolü tamamen açık kaynak kodludur ve piyasada kullanılan C++ derleyicilerinin çoğunun içerisinde bulunmaktadır.

CUDA (Compute Unified Device Architecture) [19], NVIDIA şirketinin genel maksatlı GPU kullanımı için (GPGPU) oluşturduğu bir uygulama programlama arayüzüdür (API). Bu araç sayesinde kullanıcılar C++ veya Fortran gibi üst seviye programlama dillerini kullanarak GPU donanımlarını programlayabilmektedirler. CUDA aracı aynı zamanda CPU hafızası ile harici hafızaya sahip GPU donanımlarının hafızaları arasındaki veri transferlerini de kontrollü bir şekilde yapmaya yaramaktadır. CUDA kütüphanesi içerisinde matris operasyonları için araçlar bulunmaktadır. Bu araçlara örnek

olarak CuBLAS(CUDA Basic Linear Algebra Subroutines) ve CUSPARSE( CUDA Sparse Matrix library) gösterilebilir. CUDA aracı sadece NVIDIA şirketi tarafından üretilen GPU donanımları üzerinde çalışmaktadır.

OpenCL (Open Computing Language) [38] protokolü; CPU, GPU, FPGA (Field-programmable gate arrays) gibi birbirinden farklı mimarilere sahip donanımların heterojen bir şekilde birlikte paralel çalışmasını sağlayan bir arayüzdür. OpenCL arayüzü C++ gibi programlama dillerini kullanarak bu donanımlar üzerinde çalışacak uygulama yazmak için kullanılmaktadır. OpenCL arayüzü CUDA aracından farklı olarak piyasadaki bütün GPU donanımları üzerinde çalışabilmektedir.



## 2. ÇÖZÜCÜDE KULLANILAN YÖNTEMLER

Tez dahilindeki HAD çözücüsünde kullanılan sayısal yöntemler ve HAD metotları tezin bu kısmında incelenmiştir.

### 2.1 Fiziksel Problemin Çözümünde Kullanılan Denklemler

Çözücü dahilinde kullanılan denklemler ve denklem seti bu kısımda anlatılmaktadır. Modelde kullanılan denklemlerin hepsi genel transport denklemi formatına getirilmiştir. Genel transport denklemi aşağıda görülmektedir.

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho\phi\mathbf{u}) = \nabla \cdot (\Gamma\nabla\phi) + S_\phi \quad (2.1)$$

Bu denklemde  $\frac{d(\rho\phi)}{dt}$  terimi zaman türevi,  $\nabla \cdot (\rho\phi\mathbf{u})$  terimi konveksiyon terimi,  $\nabla \cdot (\Gamma\nabla\phi)$  terimi difüzyon terimi ve  $S_\phi$  terimi kaynak terimi olarak adlandırılmaktadır.

Çözücü kapsamında çözülen denklemlerden süreklilik denklemi zamana bağlı sıkıştırılabilir olarak aşağıda gösterilmiştir.[1]

$$\frac{\partial(\rho)}{\partial t} + \nabla \cdot (\rho\mathbf{u}) = 0 \quad (2.2)$$

Süreklilik denkleminin transport değişkeninin "1" olduğu, difüzyon ve kaynak terimlerinin olmadığı görülmektedir.

Momentum denklemi sıkıştırılabilir Navier-Stokes [1] denklemlerinden oluşmaktadır.

$$\begin{aligned}\frac{\partial(\rho u_x)}{\partial t} + \nabla \cdot (\rho u_x \mathbf{u}) &= \nabla \cdot (\mu \nabla u_x) - (\nabla P)_x \\ \frac{\partial(\rho u_y)}{\partial t} + \nabla \cdot (\rho u_y \mathbf{u}) &= \nabla \cdot (\mu \nabla u_y) - (\nabla P)_y \\ \frac{\partial(\rho u_z)}{\partial t} + \nabla \cdot (\rho u_z \mathbf{u}) &= \nabla \cdot (\mu \nabla u_z) - (\nabla P)_z\end{aligned}\quad (2.3)$$

Momentum denkleminin transport değişkeninin x , y ve z eksenindeki hızlar olduğu, difüzyon katsayısının viskozite ( $\mu$ ) olduğu ve kaynak terimlerinin basınç gradyeninin negatifinin x , y ve x bileşenleri olduğu görülmektedir.

Bu çözücü, yanma problemlerinin çözümünde kullanılacaktır. Bunun ışığında çözüm uzayındaki sıcaklık farkları çok yüksek olacağı için enerji denklemi değişkeninin entalpi olmasına karar verilmiştir. Enerji denklemi aşağıdaki hali almıştır.[1]

$$\frac{\partial(\rho h)}{\partial t} + \nabla \cdot (\rho h \mathbf{u}) = \nabla \cdot \left( \frac{\mu}{\sigma_h} \nabla h + \frac{\mu}{\sigma_h} \left( \frac{\sigma_h}{Sc} - 1 \right) \sum_k h_k \nabla Y_k \right) + \frac{\partial P}{\partial t} + S_h \quad (2.4)$$

Bu denklemde  $h$  entalpiyi,  $\sigma_h$  Prandlt sayısını,  $Sc$  Schmidt sayısını ifade etmektedir. Bu denklemde  $\frac{\partial P}{\partial t}$  terimi yüksek hızlı akışlardaki basınç işinin yarattığı entalpi artışını temsil etmektedir. Denklem yeniden yazılırsa.

$$\frac{\partial(\rho h)}{\partial t} + \nabla \cdot (\rho h \mathbf{u}) = \nabla \cdot \left( \frac{\mu}{\sigma_h} \nabla h + \frac{\mu}{\sigma_h} \left( \frac{1}{Le} - 1 \right) \sum_k h_k \nabla Y_k \right) + \frac{\partial P}{\partial t} \quad (2.5)$$

Bu denklemde " $Le$ " terimi Lewis sayısını temsil etmektedir. Lewis sayısı hesaplama kolaylığı açısından bir kabul edilmiştir[1]. Bu kabulden sonra enerji denklemi aşağıdaki hali almaktadır.

$$\frac{\partial(\rho h)}{\partial t} + \nabla \cdot (\rho h \mathbf{u}) = \nabla \cdot \left( \frac{\mu}{\sigma_h} \nabla h + \right) + \frac{\partial P}{\partial t} \quad (2.6)$$

Bu denklemde görüldüğü üzere enerji denkleminin transport skaları entalpidir, difüzyon katsayısı  $\frac{\mu}{\sigma_h}$  ile ifade edilen terimdir.

Bu tez çerçevesinde incelenen akış çözücünün kimyasal reaksiyon modelleme kabiliyetine sahip olması beklenmektedir. Kimyasal reaksiyonların modellenmesi için akış çözücüsünün modelin içinde tanımlanan bütün kimyasal türler için ayrı bir kimyasal tür transport denklemi çözmesi gerekmektedir. Kimyasal tür taşıma denklemi aşağıdaki gibidir. [1]

$$\frac{\partial(\rho Y_k)}{\partial t} + \nabla \cdot (\rho Y_k \mathbf{u}) = \nabla \cdot (\rho D_k \nabla Y_k) + \dot{\omega}_k \quad (2.7)$$

Bu denklemdeki  $Y_k$  terimi  $k$  indisli kimyasal türün kütle oranını  $D_k$  terimi  $k$  indisli kimyasal türün karışım içindeki difüzyon katsayısını ifade etmektedir. Bu denklemdeki  $\dot{\omega}_k$  terimi ise kimyasal reaksiyonlardan dolayı kimyasal türdeki değişimi temsil eden kaynak terimi ifade etmektedir.

Hal denklemi olarak termodinamikte ki ideal gaz denklemi yazılıma eklenmiştir. Bu denklem Aşağıdaki şekilde gösterilmektedir.

$$P = \rho RT \quad (2.8)$$

Bu denklemden yola çıkılarak basınç düzeltme denkleminde gerekli olacak terim aşağıdaki gibi hesaplanmaktadır.

$$\left( \frac{\partial \rho}{\partial P} \right)_T = \frac{1}{RT} \quad (2.9)$$

Denklemdaki gaz sabiti  $R$ , birden çok kimyasal türün olduğu durumlarda karışımın gaz sabiti olarak hesaplanmaktadır.

Çözücü kapsamında çözülecek denklemler, transport denklemi formatında aşağıda gös-

terilmiştir.

$$\phi = \begin{pmatrix} 1 \\ u_x \\ u_y \\ u_z \\ h \\ Y_1 \\ \vdots \\ Y_n \end{pmatrix} ; \quad \Gamma = \begin{pmatrix} 0 \\ \mu \\ \mu \\ \mu \\ \frac{\mu}{\sigma_n} \\ \rho D_1 \\ \vdots \\ \rho D_n \end{pmatrix} ; \quad S_\phi = \begin{pmatrix} 0 \\ -(\nabla P)_x \\ -(\nabla P)_y \\ -(\nabla P)_z \\ \frac{\partial P}{\partial t} \\ \dot{\omega}_1 \\ \vdots \\ \dot{\omega}_n \end{pmatrix} \quad (2.10)$$

Çözücü dahilinde denklem seti seçenekleri arasında Euler denklem seti seçeneği de vardır. Euler denklem seti seçilmesi durumunda denklem setindeki viskoz terimler çözülmemektedir. Euler denklem seti seçildiği durumda çözülecek denklemlerde difüzyon terimi bulunmamaktadır.

## 2.2 Sonlu Hacimler Yöntemi

Tezin bu kısmında, 2.1 başlığı altında gösterilen denklemlerin çözüm yöntemleri ve kullanılan yöntemlerden bahsedilecektir.

Özet olarak denklem setinin çözümünde sonlu hacimler yöntemi [1] kullanılmıştır. Karmaşık geometrilerin çözümüne olanak vermek için sonlu hacimler yöntemi, genelleşmiş düzenli olmayan çözüm ağlarına uygun olarak kullanılmıştır. Çözümler sonlu hacimler hücrelerinin merkezlerinde yapılmaktadır. Genelleşmiş çözüm ağları kullanımının gerekliliği olarak sonlu hacimler yöntemi hız ve diğer değişkenleri eş konumlu olarak kullanmaktadır. Eşkonumlu çözüm ağlarında kontrol hacmi yüzeylerdeki hızlar "Rhie and Chow face interpolation" [1] yöntemi kullanılarak bulunmaktadır. Sonlu hacimler yöntemi çözümünün ihtiyaç duyduğu gradyen hesapları ise 2.8.1 başlığında ayrıntılı olarak gösterilen Green Gauss gradyen yapılandırma ile yapılmıştır.



Denklemler sistemindeki momentum denklemlerinin çözümünde kullanılan olan viskozite( $\mu$ ), Sutherland yöntemi ile sıcaklığın fonksiyonu olarak hesaplanmaktadır. Bu modelin formülasyonunu aşağıda gösterilmektedir. [11]

$$\mu = \frac{C_1 T^{3/2}}{T + C_2} \quad (2.11)$$

Bu denklemde  $C_1$  ve  $C_2$  sırasıyla Sutherland sabitlerini,  $T$  ise sıcaklığı temsil etmektedir.

Denklemler sisteminin çözümü için sonlu hacimler yöntemi kullanılmıştır. Genel transport denkleminin integral formu şu şekilde yazılabilir.

$$\int_V \frac{\partial(\rho\phi)}{\partial t} dV + \int_V \nabla \cdot (\rho\phi\mathbf{u}) dV = \int_V \nabla \cdot (\Gamma\nabla\phi) dV + \int_V S_\phi dV \quad (2.12)$$

Bu denklemin konveksiyon ve difüzyon terimlerine diverjans teoremi uygulanırsa. Bir  $\mathbf{a}$  vektörü için şu şekilde uygulanır.

$$\int_V \nabla \cdot \mathbf{a} dV = \int_A \mathbf{n} \cdot \mathbf{a} dA \quad (2.13)$$

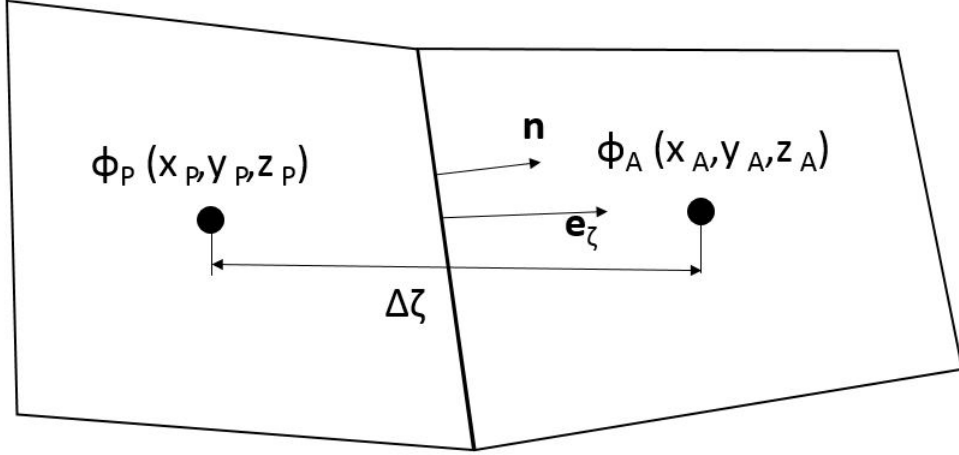
Bu denklemdeki  $\mathbf{n}$  kontrol hacmi yüzeyinin normalini,  $dV$  kontrol hacmini,  $dA$  ise kontrol hacminin yüzey alanını göstermektedir. Bu teorem integral haldeki transport denklemine uygulanırsa transport denklemi aşağıdaki hali alır.

$$\int_V \frac{\partial(\rho\phi)}{\partial t} dV + \int_A \mathbf{n} \cdot (\rho\phi\mathbf{u}) dA = \int_A \mathbf{n} \cdot (\Gamma\nabla\phi) dA + \int_V S_\phi dV \quad (2.14)$$

transport denklemi, sabit sayılı olmayan yüzlere sahip hücre merkezli eş konumlu düzgün olmayan çözüm hücreleri için yazıldığında. alan akısı içeren terimler hücrenin bütün yüzlerindeki akıların toplamı olarak yazılır.

$$\int_V \frac{\partial(\rho\phi)}{\partial t} dV + \sum \int_{A_i} \mathbf{n} \cdot (\rho\phi\mathbf{u}) dA_i = \sum \int_{A_i} \mathbf{n} \cdot (\Gamma\nabla\phi) dA_i + \int_V S_\phi dV \quad (2.15)$$

Tezin ilerleyen kısmında bu denklemdeki terimlerin ayrıştırma yöntemleri anlatılmıştır. Eş konumlu sonlu hacimler ayrıştırılması için kullanılan temel değişkenler Şekil 2.1 üzerinde gösterilmiştir [1].



Şekil 2.1: Komşu olan iki sonlu hacimler hücresinin değişkenleri

transport denklemindeki konveksiyon teriminin ayrıştırması sonlu hacimler yöntemine göre aşağıdaki gibi yapılmaktadır.[1]

$$\int_{A_i} \mathbf{n} \cdot (\rho\phi\mathbf{u}) dA_i = \mathbf{n}_i \cdot (\rho\mathbf{u}) \Delta A_i \phi_i = F_i \phi_i \quad (2.16)$$

Bu denklemdeki  $F_i$  terimi yüzey akısını  $\Delta A_i$  terimi yüzey alanını temsil etmektedir. Yüzey akısı hesabında kullanılan hız vektörü ( $\mathbf{u}$ ) hesabı eşkonumlu çözüm hücreleri kullanımından kaynaklanan dama tahtası basınç dağılımı problemi (Checkerboard Pressure Pattern) ile karşılaşmamak için "Rhie and Chow" yüzey hız interpolasyonu yöntemi ile hesaplanmaktadır. Rhie and Chow interpolasyon yöntemi kullanılan Ek Metotlar kısmında açıklanmıştır.

Konveksiyon terimi ayrıştırmasında "upwind" yöntemi kullanılmıştır bu yöntem yüzey akısının yönüne bakıp akının yönünde birinci mertebeden ayrıştırma yapmaktadır.

$$\phi_i = \begin{cases} F_i > 0 & \phi_P \\ F_i < 0 & \phi_A \end{cases} \quad (2.17)$$

Bu denklemde  $\phi_P$  terimi çözümün yapıldığı hücredeki taşınan değişkeni  $\phi_A$  terimi

ise çözümün yapıldığı yüzeyin komşuluk yaptığı hücrenin taşıma değişkenini temsil etmektedir. Çözümün ilerleyen versiyonlarında konveksiyon teriminin daha yüksek mertebeden ayrıştırmasının yapılması planlanmaktadır.

transport denklemindeki difüzyon teriminin ayrıştırmasında sonlu hacimler yönteminde merkezi farklar metodu kullanılmıştır. Difüzyon teriminin ayrıştırmasında ortogonal olmayan hücrelerin yaratacağı problemlerin önüne geçmek için kullanılması önerilen "Cross-Diffusion" terimi çözümün basitliği için şemaya katılmamıştır. Difüzyon teriminin ayrıştırması aşağıdaki gibidir. [1]

$$\int_{A_i} \mathbf{n} \cdot (\Gamma \nabla \phi) dA_i = \mathbf{n}_i \cdot (\Gamma \nabla \phi) \Delta A_i = \Gamma \left( \frac{\phi_A - \phi_P}{\Delta \zeta} \right) \Delta A_i \quad (2.18)$$

Bu denklemde  $\phi_P$  çözümün yapıldığı hücredeki taşınan değişkeni  $\phi_A$  ise çözümün ya-

pıldığı yüzeyin komşuluk yaptığı hücrenin taşıma değişkenini temsil etmektedir. Bu denklemdeki  $\Delta A_i$  terimi yüzey alanını  $\Delta \zeta$  terimi ise çözümün yapıldığı hücrenin merkezi ile çözümün yapıldığı yüzeyin komşuluk yaptığı hücre merkezi arasındaki mesafeyi temsil etmektedir.

Kaynak teriminin ayrıştırması sonlu hacimler yöntemine göre yapılmıştır.[1]

$$\int_V S_\phi dV = \bar{S} \Delta V \quad (2.19)$$

Bu denklemde  $\bar{S}$  terimi kontrol hacmindeki ortalama kaynak değişkenini  $\Delta V$  terimi ise

çözümü yapılan hücrenin hacmini ifade etmektedir.

Zaman teriminin ayrıştırması sonlu hacimler yönteminde kapalı Euler (implicit Euler) sayısal yöntemi ile yapılmıştır.

$$\int_V \frac{\partial(\rho\phi)}{\partial t} dV = \rho \left( \frac{\phi^n - \phi^{n-1}}{\Delta t} \right) \Delta V \quad (2.20)$$

Bu denklemdeki  $\Delta t$  zaman adımını  $\phi^n$  terimi çözümün yapıldığı zaman adımındaki taşınan skalerin değerini  $\phi^{n-1}$  terimi ise önceki zaman adımındaki taşınan skaler değerini temsil etmektedir.

Zamana bağlı olan problemlerde zaman adımı kullanıcıdan alınmaktadır. Zamana bağlı olmayan sıkıştırılabilir akış problemlerinde ise zaman adımı lokal zaman adımı (local timestepping) [1] yöntemi ile hesaplanmaktadır.

$$\Delta t = N_{CFL} \min \left( \frac{\Delta V}{\sum (|\mathbf{u}_i \cdot \mathbf{n}_i| + c_i) \Delta A_i}, \frac{\Delta V}{\sum \frac{\mu_i}{\rho_i} \Delta A_i} \right) \quad (2.21)$$

Bu denklemdeki  $N_{CFL}$  terimi CFL(Courant-Friedrichs-Lewy) sayısını temsil etmektedir ve kullanıcıdan girdi olarak alınmaktadır. Bu denklemdeki  $c_i$  terimi  $i$  yüzeyindeki ses hızını temsil etmektedir  $\mu_i$  terimi ise laminar ve türbülanslı viskozitenin toplamını temsil etmektedir.

Tezin başında bahsedilen 2.1'nolu denkleminin temsil ettiği 2.10'nolu denklem seti diverjans yöntemi kullanılarak 2.15'nolu denklem türetilmiştir. Bu denklemin elemanlarının teker teker ayrıştırması yapıldığında aşağıdaki denklem oluşmaktadır.

$$\rho \left( \frac{\phi_P^n - \phi_P^{n-1}}{\Delta t} \right) \Delta V + \sum \mathbf{n}_i \cdot (\rho \mathbf{u}) \Delta A_i \phi_i^n = \sum \Gamma \left( \frac{\phi_A^n - \phi_P^n}{\Delta \zeta} \right) \Delta A_i + \bar{S} \Delta V \quad (2.22)$$

Bu denklemdeki bilinmeyenler bir arada toplanıp bilinmeyenlerin çarpanları  $a$  adı verilen sabitler şeklinde yazılırsa bu denklem aşağıdaki hali alır.

$$a_P \phi_P^n + \sum a_A \phi_A^n - \left( \frac{\rho \Delta V}{\Delta t} \right) \phi_P^{n-1} + \bar{S} \Delta V = 0 \quad (2.23)$$

Bu denklemdeki bilinmeyen terimler  $\phi_P^n$  şeklinde ifade edilen çözümün yapıldığı hü-

renin çözümünün yapıldığı zaman adımındaki değeri ve  $\phi_A^n$  şeklinde ifade edilen çözümün yapıldığı hücrenin çözümünün yapıldığı yüzeyinin komşuluk oluşturduğu komşu hücrenin çözümünün yapıldığı zaman adımındaki değeridir.  $\phi_P^{n-1}$  terimi ile ifade edilen çözümün yapıldığı hücrenin önceki zaman adımındaki değeridir. Bu değer bilindiği için geri kalan kaynak terimleriyle birleştirilip  $b$  ile gösterilen bir sabit ile gösterilir. Bu işlemlerden sonra sonlu hacimler yöntemine göre ayrıştırma yapılmış transport denklemi son halini alır. transport denkleminin son hali doğrusal bir denklemdir [1].

$$a_p \phi_P^n + \sum a_A \phi_A^n + b = 0 \quad (2.24)$$

Bu lineer denklem setinin çözümünde kullanılan yöntemler tezin ilerleyen kısımlarında anlatılmıştır.

### 2.3 Basınç Merkezli Çözüm Algoritması

Fiziksel problemin çözümü için oluşturulan denklem seti sonlu hacimler yöntemi ile ayrıştırıldıktan sonra 2.24' nolu denkleme indirgenmiştir. Bu denkleminlerin çözümü için gereken algoritma için I. Demirdzic ve M. Peric tarafından geliştirilen Her Hızdaki Akışın Tahmini İçin Eş konumlu Sonlu Hacimler Yöntemi (A collocated finite volume method for predicting flows at all speeds) [5] yönteminin kullanılmasına karar verilmiştir. Bu kararın nedenlerinden biri, basınç merkezli çözüm algoritmalarının yoğunluk merkezli çözüm algoritmalarına göre daha geniş MACH sayısı aralıklarında doğru hesaplama yapmasıdır. Basınç merkezli modeller arasında bu modelin seçilmesinin nedeni ise bu modelin basınç düzeltmenin yanında yoğunluk değişkenini de düzeltiyor olmasıdır. Bu model Star-CCM+ başta olmak üzere birçok ticari yazılımda başarı ile kullanılmaktadır.

Raporun bu bölümünde iç iterasyon indisi "m" olarak zaman adımı indisi "n" olarak gösterilecektir. Çözüm algoritması kullanıcı tarafından belirlenen sayıda iç iterasyon yaptıktan sonra sonraki zaman adımına geçmektedir. Çözüm algoritmasının bir iç iterasyonda yaptığı işlemler aşağıda ayrıntılı olarak anlatılmıştır.

Çözüm algoritmasının ilk adımında sınır koşullarının değerleri belirlenir. Algoritma-

nın ikinci kısmında basınç ve hız gibi değişkenlerin gradyenleri ek metotlar kısmındaki green-gauss gradyen yapılandırma yöntemi ile yapılandırılır. Yapılandırılan basınç gradyenlerine düzeltilmemiş basınç gradyeni  $((\nabla P)^*)$  adı verilir. Yüzeylerdeki hızlar Rhie and chow interpolasyon yöntemi ile hesaplanır. Hız alanları momentum denklemleri kullanılarak çözülür.

$$a_p u_{x,P}^* + \sum a_A u_{x,A}^* - \left( \frac{\rho \Delta V}{\Delta t} \right) u_{x,P}^{n-1} + (\nabla P)_x^* = 0 \quad (2.25)$$

$$a_p u_{y,P}^* + \sum a_A u_{y,A}^* - \left( \frac{\rho \Delta V}{\Delta t} \right) u_{y,P}^{n-1} + (\nabla P)_y^* = 0 \quad (2.26)$$

$$a_p u_{z,P}^* + \sum a_A u_{z,A}^* - \left( \frac{\rho \Delta V}{\Delta t} \right) u_{z,P}^{n-1} + (\nabla P)_z^* = 0 \quad (2.27)$$

Momentum denklemlerinin ayrıştırılmasında yüzeydeki yoğunluk teriminin de hız terimi gibi ayrıştırma yöntemine göre upwind olan hücreden alınması, çözücünün sıkıştırılabilir akışları çözme amacından dolayı önem arz etmektedir. Zamana bağlı olmayan terimler okuma kolaylığı açısından "b" şeklinde gösterilirse.

$$a_p u_{x,P}^* + \sum a_A u_{x,A}^* + b_x + (\nabla P)_x^* = 0 \quad (2.28)$$

$$a_p u_{y,P}^* + \sum a_A u_{y,A}^* + b_y + (\nabla P)_y^* = 0 \quad (2.29)$$

$$a_p u_{z,P}^* + \sum a_A u_{z,A}^* + b_z + (\nabla P)_z^* = 0 \quad (2.30)$$

Bu denklemlerin sonucu çıkan hız alanı düzeltilmemiş hız alanı olarak isimlendirilmektedir. Düzeltilmemiş değişken alanları ve düzeltme terimleri aşağıdaki denklemler ile açıklanabilir.

$$\begin{aligned}
P &= P^* + P' \\
\nabla P &= \nabla P^* + \nabla P' \\
u_x &= u_x^* + u_x' \\
u_y &= u_y^* + u_y' \\
u_z &= u_z^* + u_z'
\end{aligned} \tag{2.31}$$

Yoğunluk değerinin ara değeri hesaplanmadığı için yoğunluk değeri aşağıdaki gibi önceki iterasyondaki yoğunluk değeri ile ifade edilir. [2]

$$\rho^m = \rho^{m-1} + \rho' \tag{2.32}$$

Algoritmanın sonraki adımını düzeltilmemiş hız alanları kullanılarak yüzeylerdeki düzeltilmemiş kütle akıları hesaplanmaktadır.

$$\dot{m}_f^* = \rho_f^{m-1} u_f \Delta A_f \tag{2.33}$$

Bu denklemde " $\rho_f^{m-1}$ " yüzeydeki yoğunluğun önceki iterasyondaki değerini temsil etmektedir. Yüzeydeki yoğunluk yüzeyin komşuluğunu yaptığı iki hücrenin yoğunluğunun hacimsel ağırlıklı ortalaması alınarak hesaplanabilir. Yüzeydeki yoğunluk konveksiyon teriminin hesaplanmasında kullanılan upwind metodu ile de hesaplanabilir. Bu denklemdeki " $u_f$ " terimi ise Rhie and Chow interpolasyon yöntemi ile hesaplanan yüzey hızıdır.

Çözüm algoritmasının bir sonraki adımını basınç düzeltme denklemini çözmektir. Basınç düzeltme denklemi süreklilik denkleminden türetilmektedir. Bu denklemi türetmek için ilk önce hız düzeltme denklemlerinin türetilmesi gerekmektedir. Herhangi bir eksenindeki hızın düzeltme denklemi, momentum denklemi.

$$a_p u_{x,P} + \sum a_A u_{x,A} + b_x + (\nabla P)_x = 0 \tag{2.34}$$

Ve düzeltilmemiş momentum denkleminin.

$$a_p u_{x,P}^* + \sum a_A u_{x,A}^* + b_x + (\nabla P)_x^* = 0 \quad (2.35)$$

Birbirinden çıkarılması ile türetilir.

$$a_p (u_{x,P} - u_{x,P}^*) + \sum a_A (u_{x,A} - u_{x,A}^*) + ((\nabla P)_x - (\nabla P)_x^*) = 0 \quad (2.36)$$

$$a_p u'_{x,P} + \sum a_A u'_{x,A} + (\nabla P)'_x = 0 \quad (2.37)$$

Bu denklemdaki toplam sembolü terim göz ardı edilebilir kabul edilmektedir.[5]

$$a_p u'_{x,P} = -(\nabla P)'_x \quad (2.38)$$

Bu denkleme "d" şeklinde yeni bir parametre eklenirse denklem aşağıdaki hali alır.

$$d = \frac{\Delta V}{a_p} \quad (2.39)$$

$$u'_{x,P} = -d(\nabla P)'_x \quad (2.40)$$

Bu denklem üç eksendeki hız için de yazılabilir.

$$\begin{aligned} u'_{x,P} &= -d(\nabla P)'_x \\ u'_{y,P} &= -d(\nabla P)'_y \\ u'_{z,P} &= -d(\nabla P)'_z \end{aligned} \quad (2.41)$$

Bu denklem seti aynı zamanda vektör olarak da gösterilebilir.

$$\mathbf{u}'_P = -d(\nabla P)' \quad (2.42)$$

Basınç düzeltme denkleminin türetilmesi için çözüm ağının herhangi bir yüzeyinden akan kütle akısının denklemini türetmek gerekmektedir. Problemimiz sıkıştırılabilir olduğu için bu denklem yoğunluk düzeltme terimini de içermektedir. Skalerlerin düzeltme denklemlerden yola çıkarak çözüm yapılan iterasyondaki çözüm ağının yüze-



yindeki kütle akısı aşağıdaki gibi gösterilebilir.[2]

$$\dot{m}_f^m = (\rho^{m-1} + \rho')_f (\mathbf{u}^* + \mathbf{u}')_f \Delta A_f \quad (2.43)$$

Denklemin sağ tarafı açıldığında aşağıdaki denklem elde edilir.

$$\dot{m}_f^m = (\rho_f^{m-1} u_f^* \Delta A_f) + (\rho^{m-1} \mathbf{u}' \Delta A)_f + (\mathbf{u}^* \rho' \Delta A)_f + (\rho' \mathbf{u}' \Delta A)_f \quad (2.44)$$

Bu denklemdeki son terim olan " $(\rho' \mathbf{u}' \Delta A)_f$ " terimi göz ardı edilebilir. 2.33'nolu denklem bu denklemde yerine yazılırsa ve düzeltme terimlerinin çarpımı olan terim denklemden çıkarılırsa denklem aşağıdaki hali alır.

$$\dot{m}_f^m = \dot{m}_f^* + (\rho^{m-1} \mathbf{u}' \Delta A)_f + (\mathbf{u}^* \rho' \Delta A)_f \quad (2.45)$$

Kütle akısı düzeltme denklemi bu denklemden çıkarılır.

$$\dot{m}'_f = (\rho^{m-1} \mathbf{u}' \Delta A)_f + (\mathbf{u}^* \rho' \Delta A)_f \quad (2.46)$$

Bu denklemin ilk terimi 2.42'nolu denklem yardımı ile çıkarılır.

$$(\rho^{m-1} \mathbf{u}' \Delta A)_f = -(\rho^{m-1} \Delta A_f d)(\nabla P)'_f \quad (2.47)$$

Bu denklemdeki " $d$ " terimi hız düzeltme denklemindeki terimdir, " $(\nabla P)'_f$ " terimi ise " $f$ " yüzeyindeki basınç doğrulama terimi gradyenini temsil etmektedir.

Kütle akısı doğrulama denkleminin ikinci terimi ise termodinamik yasaları yardımıyla bulunmaktadır.

$$\frac{\rho'}{P'} \approx \left( \frac{\partial \rho}{\partial P} \right)_T \quad (2.48)$$

Bu denklemde " $\left( \frac{\partial \rho}{\partial P} \right)_T$ " terimi " $T$ " sıcaklığındaki yoğunluğun basınca oranını temsil etmektedir. Çözüm için seçilen hal denklemi ile hesaplanmaktadır. Bu denklemden yola çıkılarak yoğunluk düzeltme terimi aşağıdaki gibi hesaplanmaktadır.

$$\rho' = \left( \frac{\partial \rho}{\partial P} \right)_T P' \quad (2.49)$$

Bu terim yerine yazılırsa.

$$(\mathbf{u}^* \rho' \Delta A)_f = \mathbf{u}^* \Delta A \left( \frac{\partial \rho}{\partial P} \right)_T P' \quad (2.50)$$

Düzeltilmemiş hız terimi algoritmanın önceki adımlarında hesaplanan düzeltilmemiş kütle akısı terimi şeklinde yazılırsa.

$$(\mathbf{u}^* \rho' \Delta A)_f = \frac{\dot{m}_f^*}{\rho^{m-1}} \left( \frac{\partial \rho}{\partial P} \right)_T P' \quad (2.51)$$

Çıkarılan bu ki terim, kütle akısı düzeltme denkleminde yerine yazılırsa kütle akısı düzeltme denklemi aşağıdaki hali alır.

$$\dot{m}'_f = -(\rho^{m-1} \Delta A_f d)(\nabla P)'_f + \frac{\dot{m}_f^*}{\rho^{m-1}} \left( \frac{\partial \rho}{\partial P} \right)_T P' \quad (2.52)$$

Basınç düzeltme denkleminin çıkarımı süreklilik ve momentum denklemlerine dayanmaktadır. Süreklilik denklemi.

$$\frac{(\rho - \rho^{n-1})\Delta V}{\Delta t} \sum \rho \mathbf{u}_f \Delta A_f = 0 \quad (2.53)$$

Bu denklemde yüzeylerdeki kütle akıları yerine yazılırsa.

$$\frac{(\rho - \rho^{n-1})\Delta V}{\Delta t} \sum \dot{m}_f = 0 \quad (2.54)$$

Yoğunluk terimi ve kütle akısı terimleri düzeltilmemiş terimler ve düzeltme terimleri şeklinde bileşenlerine ayrılırsa bu denklem aşağıdaki hali alır.

$$\frac{((\rho^{m-1} + \rho') - \rho^{n-1})\Delta V}{\Delta t} \sum (\dot{m}_f^* + \dot{m}'_f) = 0 \quad (2.55)$$

Bu denklem düzenlenerek yeniden yazılırsa.

$$\frac{\rho' \Delta V}{\Delta t} + \sum \dot{m}'_f + \frac{(\rho^{m-1} - \rho^{n-1}) \Delta V}{\Delta t} + \sum \dot{m}^*_f = 0 \quad (2.56)$$

Bu denklemde bilinen terimler ayrı bir şekilde yazılırsa.

$$\frac{(\rho^{m-1} - \rho^{n-1}) \Delta V}{\Delta t} + \sum \dot{m}^*_f = Q_m \quad (2.57)$$

Bu denklemdeki " $Q_m$ " terimi düzeltilmemiş süreklilik denkleminde düzeltilmesi gereken dengesizliği temsil eder. Bu dengesizliğin hesabı için gerekli olan bütün değişkenler algoritmanın bu kısmına gelindiğinde hesaplanmıştır. Süreklilik denklemi bu dengesizlik terimi kullanılarak yeniden yazılırsa aşağıdaki hali alır.

$$\frac{\rho' \Delta V}{\Delta t} + \sum \dot{m}'_f + Q_m = 0 \quad (2.58)$$

Bu denklemdeki kütle akısı düzeltme terimleri yerine 2.52'nolu denklemdeki basınç düzeltme terimlerini içeren ifade yazılırsa bu denklem aşağıdaki hali alır.

$$\frac{\rho' \Delta V}{\Delta t} + \sum_{f=1}^{n_f} -(\rho^{m-1} \Delta A_f d) (\nabla P)'_f + \frac{\dot{m}^*_f}{\rho^{m-1}} \left( \frac{\partial \rho}{\partial P} \right)_T P' + Q_m = 0 \quad (2.59)$$

Bu denklemdeki basınç düzeltme terimleri haricindeki bütün terimler bilinmektedir. Bu denklemin amacı basınç düzeltme teriminin her hücre için bulunmasıdır. Bu modeldeki basınç düzeltme denklemi SIMPLE [6] algoritmasındaki basınç düzeltme teriminden farklıdır bu yüzden SIMPLE algoritmasındaki kullanılan denklem çözüm metodu işe yaramamaktadır. Demirtzic [5] makalesinde bu denklemdeki basınç düzeltme gradyeni terimlerinin yaklaşık değerlerinin bulunmasının ve bunlarla çözüm yapılmasının doğru sonuç verdiğini söylemektedir. Bu yazılımın içinde de bu denklemler, basınç düzeltme gradyeni değerlerinin önceki iterasyondan alınması ile çözülmektedir. Gradyen terimleri bilindiğinde basınç düzeltme denklemi kapalı olarak matris çözücüsü ile çözülebilmektedir.

Çözüm algoritması basınç düzeltme terimlerini hesapladıktan sonra basınçları düzeltmektedir.

$$P = P^* + \omega_p P' \quad (2.60)$$

Bu denklemdeki " $\omega_p$ " terimi basınç rahatlatma katsayısını temsil etmektedir.

Çözüm algoritması sonraki adım olarak sınır koşullarındaki basınç düzeltme değişkenlerini güncellemektedir. Çözüm algoritmasının sonraki adımını ise kütle akılarını düzeltmedir.

$$\dot{m}_f = \dot{m}_f^* + \dot{m}'_f \quad (2.61)$$

Çözüm algoritması sonraki adım olarak hücrelerdeki hız değerlerini düzeltmektedir. Hücrelerdeki hız terimlerinin düzeltilme işlemi 2.42'nolu denklem ile hesaplanan değer kullanılarak aşağıdaki gibi yapılmaktadır.

$$\mathbf{u} = \mathbf{u}^* + \omega_u \mathbf{u}' \quad (2.62)$$

Bu denklemdeki " $\omega_u$ " terimi hız rahatlatma katsayısını temsil etmektedir.

Çözüm algoritmasının sonraki adımında yoğunluk değerleri değişen basınç değerleri kullanılarak hal denklemi vasıtasıyla güncellenmektedir. Çözüm algoritması son adım olarak eğer akışın içinde kimyasal denklem varsa reaksiyon hızlarını hesaplayıp düzeltilmiş basınç yoğunluk ve hız değerleri ile enerji denklemini ve kimyasal tür transport denklemlerini çözmektedir.

## 2.4 Türbülansın Modellenmesi

Türbülanslı akış problemlerinde türbülansın akış üzerindeki etkilerini dikkate almak üzere RANS(Reynolds Averaged Navier-Stokes) denklemleri çözülmüştür. RANS türbülans modeli olarak  $k - \varepsilon$  [9] kullanılmıştır. RANS denklem sistemi, Navier-Stokes denklemlerinin değişkenlerinin zaman ortalaması alınarak oluşturulur. Zaman ortalaması yönteminde her skalar büyüklüğün anlık değeri, zaman ortalamalı değeri ile çalkantılı değerinin toplamına eşittir.

$$\phi = \bar{\phi} + \phi' \quad (2.63)$$

Bu denklemdaki zaman ortalaması terimi aşağıdaki denkleme dayanmaktadır.

$$\bar{\phi} = \frac{1}{\Delta t} \int_0^{\Delta t} \phi(t) dt \quad (2.64)$$

Navier-Stokes denklemlerinin zaman ortalaması alınmış hali aşağıdaki gibidir.

$$\begin{aligned} \frac{\partial(\bar{\rho}u_x)}{\partial t} + \nabla \cdot (\bar{\rho}u_x \mathbf{u}) &= \nabla \cdot (\mu \nabla \bar{u}_x) - (\nabla \bar{P})_x \\ &+ \left[ -\frac{\partial(\bar{\rho}u'_x u'_x)}{\partial x} - \frac{\partial(\bar{\rho}u'_y u'_x)}{\partial y} - \frac{\partial(\bar{\rho}u'_z u'_x)}{\partial z} \right] \\ \frac{\partial(\bar{\rho}u_y)}{\partial t} + \nabla \cdot (\bar{\rho}u_y \mathbf{u}) &= \nabla \cdot (\mu \nabla \bar{u}_y) - (\nabla \bar{P})_y \\ &+ \left[ -\frac{\partial(\bar{\rho}u'_x u'_y)}{\partial x} - \frac{\partial(\bar{\rho}u'_y u'_y)}{\partial y} - \frac{\partial(\bar{\rho}u'_z u'_y)}{\partial z} \right] \\ \frac{\partial(\bar{\rho}u_z)}{\partial t} + \nabla \cdot (\bar{\rho}u_z \mathbf{u}) &= \nabla \cdot (\mu \nabla \bar{u}_z) - (\nabla \bar{P})_z \\ &+ \left[ -\frac{\partial(\bar{\rho}u'_x u'_z)}{\partial x} - \frac{\partial(\bar{\rho}u'_y u'_z)}{\partial y} - \frac{\partial(\bar{\rho}u'_z u'_z)}{\partial z} \right] \end{aligned} \quad (2.65)$$

RANS sonucu ortaya çıkan türbülanslı gerilmeler Boussinesq yaklaşımı kullanılarak şu şekilde ifade edilmiştir [1].

$$\tau_{ij} = \overline{\rho u'_i u'_j} = \mu_t \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (2.66)$$

Boussinesq yaklaşımı zaman ortalamalı momentum denklemlerine uygulandığında momentum denklemleri aşağıdaki hali almaktadır.

$$\begin{aligned}\frac{\partial(\overline{\rho u_x})}{\partial t} + \nabla \cdot (\overline{\rho u_x \mathbf{u}}) &= \nabla \cdot ((\mu + \mu_t) \nabla \overline{u_x}) - (\nabla \overline{P})_x \\ \frac{\partial(\overline{\rho u_y})}{\partial t} + \nabla \cdot (\overline{\rho u_y \mathbf{u}}) &= \nabla \cdot ((\mu + \mu_t) \nabla \overline{u_y}) - (\nabla \overline{P})_y \\ \frac{\partial(\overline{\rho u_z})}{\partial t} + \nabla \cdot (\overline{\rho u_z \mathbf{u}}) &= \nabla \cdot ((\mu + \mu_t) \nabla \overline{u_z}) - (\nabla \overline{P})_z\end{aligned}\quad (2.67)$$

Denklemlerdeki  $\mu_t$  terimi türbülanslı viskoziteyi temsil etmektedir.

Çözülen denklem sistemindeki momentum dışındaki denklemlerin de zaman ortalaması alınmaktadır. Temel transport denkleminin zaman ortalaması alınmış ve Boussinesq yaklaşımı uygulanmış formu aşağıdaki gibidir.

$$\frac{\partial(\overline{\rho \phi})}{\partial t} + \nabla \cdot (\overline{\rho \phi \mathbf{u}}) = \nabla \cdot ((\Gamma + \Gamma_t) \nabla \overline{\phi}) + \overline{S_\phi} \quad (2.68)$$

Bu denklemdeki  $\Gamma_t$  terimi türbülanslı difüzyon katsayısını temsil etmektedir ve aşağıda verildiği gibi hesaplanmaktadır.

$$\Gamma_t = \frac{\mu_t}{\sigma_t} \quad (2.69)$$

Literatürde görüldüğü üzere bu oranın her durumda sabit kabul edilebileceği deneylerle kanıtlanmıştır [1] [2]. Denklemdeki  $\sigma_t$  terimi entalpi denkleminde türbülanslı Prandtl sayısı olarak, kimyasal tür denkleminde ise türbülanslı Schmidt sayısı olarak geçmektedir. Denklemlerde görüldüğü üzere türbülanslı viskozite ( $\mu_t$ ) terimi haricindeki bütün terimlerin çözüm yöntemleri önceki başlıklarda gösterilmektedir.  $k - \varepsilon$  türbülans modelinin ana amacı " $\mu_t$ " değişkeninin değerini bulmaktır.

$k - \varepsilon$  modeli iki adet transport denklemi çözmektedir. Bu denklemlerden ilki türbülans kinetik enerjisi ( $k$ ) için, ikincisi ise türbülanslı kinetik enerjinin yitimi ( $\varepsilon$ ) için çözülmektedir. Türbülans kinetik enerjisi için transport denklemi aşağıdaki gibidir [2].

$$\frac{\partial(\rho k)}{\partial t} + \nabla \cdot (\rho k \mathbf{u}) = \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right) + 2\mu_t S_{ij} \cdot S_{ij} - \rho \varepsilon \quad (2.70)$$

Denklemdeki pozitif kaynak terimi, türbülans kinetik enerjisinin üretim hızını temsil

Çizelge 2.1:  $k - \varepsilon$  türbülans modeli sabitleri

$C_\mu$	$\sigma_k$	$\sigma_\varepsilon$	$C_{1\varepsilon}$	$C_{2\varepsilon}$
0.09	1	1.3	1.44	1.92

etmektedir. Denklemdaki negatif kaynak terimi ise türbülans kinetik enerjisinin yitim hızını temsil etmektedir. Denklemdaki " $\sigma_k$ " terimi modele özgü bir sabittir. Türbülans kinetik enerjisinin yitimi aşağıdaki transport denklemi ile hesaplanmaktadır [2].

$$\frac{\partial(\rho\varepsilon)}{\partial t} + \nabla \cdot (\rho\varepsilon\mathbf{u}) = \nabla \cdot \left( \frac{\mu_t}{\sigma_\varepsilon} \nabla \varepsilon \right) + C_{1\varepsilon} \frac{\varepsilon}{k} 2\mu_t S_{ij} \cdot S_{ij} - C_{2\varepsilon} \rho \frac{\varepsilon^2}{k} \quad (2.71)$$

Denklemdaki pozitif kaynak terimi  $\varepsilon$  değişkeninin oluşum hızını, negatif kaynak terimi ise  $\varepsilon$  değişkeninin yitim hızını temsil etmektedir. Denklemdaki  $S_{ij}$  terimi hücrenin gerinim tensörüdür. Denklemdaki  $\sigma_\varepsilon$ ,  $C_{1\varepsilon}$  ve  $C_{2\varepsilon}$  terimleri modele özgü sabitlerdir. Türbülanslı vizkozite  $k$  ve  $\varepsilon$  değişkenleri kullanılarak aşağıdaki gibi hesaplanmaktadır.

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \quad (2.72)$$

Bu denklemde  $\rho$  yoğunluğu,  $C_\mu$  modele özgü sabiti temsil etmektedir. Modele özgü sabitlerin değerleri Çizelge 2.1 'de verilmiştir [1].

Çözücüde RANS denklemleri ile " $k - \varepsilon$ " modeli seçildiğinde çözücünün çözdüğü transport denklem seti aşağıdaki hali almaktadır.

Bu denklem seti zamana bağlı RANS ve zamana bağlı olmayan RANS denklemleri için çözülebilmektedir. " $k$ " ve " $\varepsilon$ " denklemleri çözüm algoritmasının içinde basınç düzeltme denkleminde sonra ve entalpi denkleminde önce çözülmektedir.

$$\phi = \begin{pmatrix} 1 \\ \bar{u}_x \\ \bar{u}_y \\ \bar{u}_z \\ k \\ \varepsilon \\ \bar{h} \\ \bar{Y}_1 \\ \vdots \\ \bar{Y}_n \end{pmatrix} ; \quad \Gamma = \begin{pmatrix} 0 \\ \mu + \mu_t \\ \mu + \mu_t \\ \mu + \mu_t \\ \mu + \frac{\mu_t}{\sigma_k} \\ \frac{\mu_t}{\sigma_e} \\ \frac{\mu}{\sigma_h} + \frac{\mu_t}{\sigma_{th}} \\ \rho D_1 + \frac{\mu_t}{\sigma_{ty}} \\ \vdots \\ \rho D_n + \frac{\mu_t}{\sigma_{ty}} \end{pmatrix} ; \quad S_\phi = \begin{pmatrix} 0 \\ -(\nabla \bar{P})_x \\ -(\nabla \bar{P})_y \\ -(\nabla \bar{P})_z \\ 2\mu_t S_{ij} \cdot S_{ij} - \rho \varepsilon \\ C_{1\varepsilon} \frac{\varepsilon}{k} 2\mu_t S_{ij} \cdot S_{ij} - C_{2\varepsilon} \rho \frac{\varepsilon^2}{k} \\ 0 \\ \dot{\omega}_1 \\ \vdots \\ \dot{\omega}_n \end{pmatrix} \quad (2.73)$$

## 2.5 Kimyasal Reaksiyonların Modellenmesi

Bu kısımda kimyasal reaksiyonların modellenme yöntemleri incelenmiştir. Kimyasal reaksiyonların kimyasal tür taşıma denklemlerinde yarattığı kaynak terimlerin hesaplama yöntemleri gösterilmiştir. Tezin bu kısmında kafa karışıklığını önlemek için bu bölümdeki denklemlerde "m" terimi reaksiyon indisi olarak, "n" terimi ise kimyasal tür indisi olarak kullanılmıştır. Buradan yola çıkarak " $n_m$ " terimi toplam reaksiyon sayısını, " $n_n$ " terimi toplam kimyasal tür sayısını ifade etmektedir.

Kimyasal reaksiyonları modellemek için Arrhenius yaklaşımı kullanılmıştır.[4]

$$RR_m = A_p e^{\left(\frac{-E_A}{RT}\right)} \prod [C_n]^{\psi_n} \quad (2.74)$$

Bu modelde  $A_p$  reaksiyonun ön faktörünü (pre-exponential factor),  $E_A$  reaksiyonun aktivasyon enerjisini,  $R$  evrensel gaz sabitini,  $T$  karışımın sıcaklığını,  $C_n$ ,  $n$  indisli kimyasal türün karışımındaki molar konsantrasyonunu,  $\psi_n$  ise  $n$  indisli kimyasal türün reaksiyondaki değişim faktörünü (rate exponent) temsil etmektedir. Bu modelin amacı  $RR_m$



terimi ile temsil edilen  $m$  indisli reaksiyonun reaksiyon hızını bulmaktır.

Kimyasal tür transport denklemindeki kaynak terimin hesabı ise aşağıdaki denklem ile yapılmaktadır.

$$\dot{\omega}_n = M_n \sum_{m=1}^{n_m} \dot{q}_{m,n} \quad (2.75)$$

Denklemdaki  $M_n$ ,  $n$  indisli kimyasal türün molar kütleini ifade etmektedir.  $\dot{q}_{m,n}$  ise aşağıdaki gibi ifade edilmektedir.

$$\dot{q}_{m,n} = (v''_{n,m} - v'_{n,m})RR_m \quad (2.76)$$

Entalpi değerleri kullanılarak sıcaklık değeri her iterasyonda her hücre için şu şekilde hesaplanmaktadır.  $v'_{n,m}$  ve  $v''_{n,m}$  reaksiyonun sırasıyla ürün ve reaktantlarının stokiyo-metrik sabitlerini göstermektedir.

$$T = \frac{h - \sum_{n=1}^{n_n} Y_n h_{f,n}^0 - \left(\frac{|u|^2}{2}\right)}{c_{p,mix}} \quad (2.77)$$

Bu denklemda  $c_{p,mix}$  değişkeni karışımın sabit basınçtaki özgül ısısını temsil etmektedir. Karışımın özgül ısısı kimyasal türlerin özgül ısılarından aşağıdaki formül ile hesaplanmaktadır.

$$c_{p,mix} = \sum_{n=1}^{n_n} Y_n c_{p,n} \quad (2.78)$$

Bu denklem karışımın özgül ısısının değerini kimyasal türlerin özgül ısılarının kimyasal türlerin kütle oranları ile ağırlıklı ortalamalarını alarak hesaplanmaktadır. Bu yöntem ile bütün termodinamik değişkenlerin karışım değerlerini hesaplamakta kullanılmaktadır.

Çözücüde kullanılan hal denklemine kiyasal reaksiyonlu termal ideal gazlar ismi verilmektedir [1]. Kimyasal türlerin özgül ısıları  $c_{p,n}$  NASA veri tabanından yedinci mer-tebe polinomlar kullanılarak hesaplanmaktadır.

$$c_p = c_1 + c_2T + c_3T^2 + c_4T^3 + c_5T^4 + c_6T^5 + c_7T^6 \quad (2.79)$$

Bu denklemde  $c$  katsayıları NASA veri tabanındaki katsayıları,  $T$  sıcaklığı temsil etmektedir. Bu yöntem ile referans sıcaklıktaki oluşum entalpisi değerleri de aynı veri tabanı yardımı ile hesaplanmaktadır. Çözücü NASA termodinamik veri tabanı formunda 2 farklı aralıkta  $c_p$  değişkeni alabilmektedir. Bu aralıkların düşük olanı 300 K ile 1000 K aralığında, yüksek olanı ise 1000 K ile 6000 kelvin aralığındadır.

Bazen kimyasal reaksiyonlardan gelen kaynak terimler problemde kullanılan sayısal yöntemin kararsız olmasına yol açabilmektedirler. Bu kararsızlığın önüne geçmek için çözüm doğruluğundan feragat edilip kaynak terim limitlemesi yapılabilmektedir. kaynak terim limitlemesinin çalışma prensibinde, Reaksiyon hızı her hücre için hesaplanırken reaksiyon hızından gelen kaynak teriminin o hücrenin içindeki yakıtın kütleli oranını o zaman aralığı için sıfırın altına indirip indirmediği ölçülür. Eğer sıfırın altına indiriyor ise reaksiyon hızı yakıtın kütleli oranı sıfır olacak şekilde limitlenir.

## 2.6 Sınır Koşullarının Uygulanması

Çözücü için 4 temel sınır koşulu kodlanmıştır. Bu sınır koşulları; hız girişi, duvar, basınç çıkışı ve boş sınır koşulu olarak adlandırılmaktadır.

Hız girişi sınır koşulunda akışın bütün özellikleri kullanıcıdan alınmaktadır ve çözüm ağının dışında konumlandığı varsayılan hayalet hücreye zorlanılmaktadır. Entalpi denkleminin transport değişkenleri yerine kullanıcıdan sıcaklık değeri alınmaktadır. Hayalet hücrenin entalpisi kullanıcıdan alınan sıcaklık kullanılarak aşağıdaki gibi hesaplanmaktadır.

$$h = T c_{p,mix} + \left( \frac{|\mathbf{u}|^2}{2} \right) \quad (2.80)$$

Aynı entalpi denkleminde olduğu gibi türbülans denklemlerinde de transport katsayıları kullanıcıdan doğrudan alınmamaktadır. Türbülans değişkenleri kullanıcıdan türbülans şiddeti ( $T_u$ ) ve türbülans uzunluk ölçeği ( $\ell$ ) olarak alınmaktadır. Sınırdaki türbülans kinetik enerjisi aşağıdaki denklem ile hesaplanmaktadır.

$$k = \frac{2}{3}(U_{ref}T_u)^2 \quad (2.81)$$

Bu denklemdeki  $U_{ref}$  sınırdaki referans hızı temsil etmektedir. Türbülans kinetik enerjisinin yitimi değişkeni sınırdaki aşağıdaki denklem ile hesaplanmaktadır.

$$\varepsilon = C_\mu^{3/2} \frac{k^{3/2}}{\ell} \quad (2.82)$$

Literatürde  $\ell$  değeri genellikle hidrolik çapın küçük bir yüzdesi olarak alınır.

Basınç çıkışı sınır koşulunda ise hücrenin yüzeyine kullanıcıdan alınan basınç değeri zorlanır, diğer bütün değişkenlerin yüzeydeki değerleri sınır koşulunun bulunduğu yüzeye sahip olan hücreden alınır.

Duvar sınır koşulu, kaymalı ve kaymasız olmak üzere iki farklı şekilde uygulanmaktadır. Kaymasız duvar koşulu Euler denklem sistemi çözülürken problemde bulunan bütün duvarlara veya Diğer denklem sistemleri çözülürken problemde bulunan ve kullanıcı tarafından belirtilen duvarlara uygulanır.

İki duvar koşulunda da sınır koşulunun olduğu yüzeyin normalinde hız yoktur dolayısıyla yüzeyden konveksiyon yolu ile hiçbir transport skalarının akışı taşınmaz. Kaymalı duvar koşulunda yüzey ile aynı düzlemde hız bileşeni olabilmektedir. Kaymasız duvar sınır koşulunda ise sınırın olduğu yüzeyde hiçbir şekilde hız oluşmamaktadır. Entalpi ve kimyasal tür transport denklemleri duvar sınır koşullarında hiçbir akı hesaplanmamaktadır. Çözücüde duvar koşulları sadece adyabatik olarak uygulanmaktadır. Duvar sınır koşulunda türbülans kinetik enerjisi sıfır olarak alınmaktadır. Türbülans kinetik enerjisinin yitimi ise aşağıdaki denklem ile hesaplanmaktadır [2].

$$\varepsilon = \nu \left( \frac{\partial^2 k}{\partial n^2} \right)_{duvar} \quad (2.83)$$

Bu denklemde  $\nu$  kinematik viskoziteyi temsil etmektedir.

Çözücüdeki boş sınır koşulu iki boyutlu analizlerin yapılabilmesini sağlamak için eklenmiştir. Çözücüde iki boyutlu çözüm, üç boyutlu ve bir hücre kalınlığında çözüm ağı

oluşturup çözüm ağının arka ve ön yüzü boş sınır koşulu olarak seçilerek sağlanmaktadır.

## 2.7 Doğrusal Denklem Sistemi Çözümü için Kullanılan Metodlar

Tez dahilindeki çözücününün zamandaki ayrıştırması kapalı(implicit) olarak yapılmıştır. Bu yüzden denklem sistemi doğrusal hale getirildikten sonra doğrusal denklem sistemini çözecek bir çözücüye ihtiyaç duyulmaktadır. Literatürde görüldüğü üzere bu boyutlardaki seyrek matrisler genellikle iteratif çözücülerle çözülmektedir [2]. Tez dahilindeki çözücüdeki matris çözme algoritmalarının hepsi bu amaçla iteratif olarak seçilmiştir. Akış çözücüsü dahilinde doğrusal denklem sistemi çözümü için iki alternatif vardır. Alternatiflerden biri akış çözücüsünün içine gömülü şekilde kodlanmış Successive Over-Relaxation matris çözücüsüdür. Diğer alternatif ise üçüncü parti olarak çözücüye eklenmiş ViennaCL isimli matris çözme kütüphanesidir [13]. ViennaCL alternatifinde AMG metodu ile birlikte Conjugate Gradient matris çözücüsü kullanılmaktadır. Bu yöntemler aşağıda incelenmiştir.

### 2.7.1 Successive over-relaxation Matris Çözme Metodu

Peric'in yazmış olduğu metodun kapalı(Implicit) zaman ayrıştırması kullanılabilmesi için bir matris çözücüsüne ihtiyacı vardır. Yapısal olmayan ağların kullanılmasından dolayı bu metodun içinde çok sayıda "0" değişkeni bulunduran matrislerde çalışması gerekmektedir. Sayısal metodun ıraksamaması için bu matris çözücüsünden rahatlama katsayısı kullanması beklenmektedir. Bu beklentilere uyan matris çözücüsü Successive over-relaxation metodu seçilmiştir. Bu metod Gauss-Seidel metodunun rahatlama katsayısı bulunduran halidir.

Successive over-relaxation metodunda en genel çerçevede " $Ax = b$ " şeklinde bir doğrusal denklem sistemini çözer. Bu yöntemin formülasyonu aşağıdaki gibidir [8].

$$x_i^{k+1} = (1 - \omega)x_i^k + \frac{\omega}{a_{ii}} \left( b_i - \sum_{j<i} a_{ij}x_j^{k+1} - \sum_{j>i} a_{ij}x_j^k \right) \quad (2.84)$$

Bu denklemde " $x_i^k$ ", " $x$ " dizisinin " $i$ " elemanının " $k$ " iterasyonundaki değeridir, " $a_{ij}$ "

ise " $A$ " matrisinin " $i$ " satırı ve " $j$ " sütunundaki elemanını temsil eder. Denklemdaki " $b_i$ " terimi " $b$ " matrisinin " $i$ " indisli elemanıdır. Bu denklemdaki " $\omega$ " terimi ise her yeni iterasyonun rahatlama katsayısıdır.

### 2.7.2 Algebraic Multigrid metodu

Algebraic Multigrid metodu, sonlu elemanlar ve HAD denklem sistemlerinin yakınsama hızını arttırmak için kullanılan multigrid [15] metotlarının doğrusal denklem sistemi çözümünde kullanılmasıdır. Tez dahilinde bahsedilen akış çözücüsünde kullanılan sonraki başlıkta bahsedilen Conjugate Gradient matris çözme metodu ile birlikte kullanılmaktadır. Algebraic Multigrid, Conjugate Gradient matris çözme metodu gibi üçüncü parti yazılım olan ViennaCL kütüphanesinden alınmıştır [13]. Çözücü içindeki Algebraic Multigrid metodu Smoothed Aggregation yöntemi ile kullanılmaktadır. AMG metodu kaba matristeki  $e$  değerini hesaplama adımı ile başlar [16].

$$Ae = r \quad (2.85)$$

Bu denklemdaki  $r$  değişkeni aşağıdaki gibidir.

$$r = Ax - b \quad (2.86)$$

$e$  değeri kaba matriste hesaplandıktan sonra ince matrisi düzeltmek için kullanılır.

$$x \leftarrow x + e \quad (2.87)$$

AMG metodundaki Smoothed Aggregation matris kabalaştırma işlemi bütün matris elemanlarının kontrol edilmesi ile başlar. Eğer matris elemanı aşağıdaki koşulu destekliyorsa kaba matris hesabına katılır.

$$|a_{i,j}| > \theta \sqrt{|a_{i,i}a_{j,j}|} \quad (2.88)$$

Bu denklemdaki  $\theta$ , etki parametresinin eşik değeridir (Influence treshold) ve 0 ile 1 arasında kullanıcıdan alınan bir değerdir. AMG metodunda kaba matristen ince matrise

geçiş işlemi interpolasyon (Interpolation, Prolongation) yöntemi ile yapılır.

$$e_i = \sum w_{i,j} e_j \quad (2.89)$$

Bu denklemdaki  $w$  değişkenleri intepolasyonda kullanılan ağırlık parametreleridir. Bu ağırlık parametreleri aggregation yöntemi kullanıldığında matrisin  $j$  elemanı kök indislerden biri ise 1 değilse 0 olarak alınır. Smoothed aggregation yöntemi uygulandığında ise bu değer yumuşatılarak uygulanır [17].

### 2.7.3 Conjugate Gradient Matris çözme metodu

Conjugate Gradient metodu çözücü içinde AMG ile ViennaCL kütüphanesi tarafından kullanılmaktadır [13]. Conjugate Gradient algoritması  $Ax = b$  problemini  $x_0$  başlangıç koşulu ile çözmektedir. Algoritma öncelikle iterasyon değişkenlerinin başlangıç koşullarını oluşturur [14].

$$\mathbf{p}_0 = \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0 \quad ; \quad k = 0 \quad (2.90)$$

Algoritma sonraki aşamada  $\mathbf{r}_{k+1}$  değeri önceden belirlenen bir değer altına inene kadar aşağıdaki denklemleri çözer [14].

$$\begin{aligned} \alpha_k &= \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k} \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{p}_k \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k \\ \beta_k &= \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k} \\ \mathbf{p}_{k+1} &= \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \\ k &= k + 1 \end{aligned} \quad (2.91)$$

İterasyon sayısı limiti veya  $\mathbf{r}_{k+1}$  değerinin limit koşulu sağlandığında, algoritma  $\mathbf{x}_{k+1}$  değerini sonuç olarak döndürmektedir.

## 2.8 Kullanılan Ek Metotlar

Bu tezde anlatılan akış çözücüsünün kullandığı diğer metotlar bu kısımda raporlanıp incelenmiştir.

### 2.8.1 Gauss Green Gradyan Yapılandırma Metodu

HAD çözücüsünün içinde skalar büyüklüklerin gradyanlarının kullanılması gerekmektedir. Bu gradyanlarının her hücrenin merkezinde depolanan skalar büyüklükler kullanılarak hesaplanması gerekmektedir. Gradyan hesaplamak için literatürde birçok yöntem bulunmaktadır. Bu çözücüde Gauss Green metodu kullanılmaktadır.

Gauss Green gradyan yapılandırma metodu diverjans teoreminden[1] türetilmiştir. diverjans teoremi:

$$\int_V \nabla \phi dV = \int_A \phi dA \quad (2.92)$$

Yukarıdaki denklem ayrıştırıldığında sonlu hacimler yönteminde bir gradyan yapılandırma metodu olarak kullanılabilir.

$$(\nabla \phi)_i^n = \frac{1}{V} \sum_f \phi_f A_f \quad (2.93)$$

Yukarıdaki denklemde "f" yüzeyleri belirtmektedir. Gradyenin hesaplandığı skalar değişkenin yüzey değeri aşağıdaki gibi ortalama alınarak bulunur.

$$\phi_f = \frac{\phi_i + \phi_j}{2} \quad (2.94)$$

### 2.8.2 Rhie and Chow Yüzey İnterpolasyonu Metodu

Rhie and Chow [1] yüzey interpolasyonu metodu eş konumlu olmayan çözüm ağlarında yüzeylerdeki hız değerlerinin hücre merkezlerindeki hız değerleri, basınç değerleri ve basınç gradyeni değerleri ile hesaplanması şeklinde çalışır. Rhie and Chow interpolasyonu metodunun lineer interpolasyon yerine kullanılmasının sebebi, lineer interpolasyon ile yüzey hızı hesaplanan durumlarda "checkerboard pressure field" adlı probleme rastlanmasıdır. Bu problem, basınç eş eğrilerinin dalgali şekilde oluşması

şeklinde ortaya çıkmaktadır.

Rhie and Chow interpolasyon metodu aşağıdaki gibidir.

$$u_f = \left( \frac{\mathbf{u}_P + \mathbf{u}_A}{2} \right) \cdot \mathbf{n} + \frac{1}{2} \left( \frac{\Delta V_P}{a_P} + \frac{\Delta V_A}{a_A} \right) \left( \frac{p_P - p_A}{\Delta \zeta} \right) - \frac{1}{2} \left( \frac{\Delta V_P}{a_A} (\nabla P)_P + \frac{\Delta V_A}{a_A} (\nabla P)_A \right) \cdot \mathbf{e}_\zeta \quad (2.95)$$

Bu denklemde " $\mathbf{u}_P$ " ve " $\mathbf{u}_A$ " terimleri hesap yapılan ve komşu hücredeki hız vektörlerini temsil etmektedir. " $P$ " indisli terimler hesap yapılan hücreyi, " $A$ " indisli terimler komşu hücreyi temsil etmektedir. Bu denklemde " $\Delta V$ ", " $(\nabla P)$ " ise basınç gradyenini temsil etmektedir. Bu denklemdeki " $\Delta \zeta$ " terimi hücre merkezleri arasındaki mesafeyi göstermektedir. Bu denklemdeki " $\mathbf{e}_\zeta$ " ise hücre merkezlerinin birbirlerine göre konumunu temsil eden birim vektörü temsil etmektedir. " $\mathbf{e}_\zeta$ " terimi görsel olarak Şekil 2.1'de görülebilmektedir.



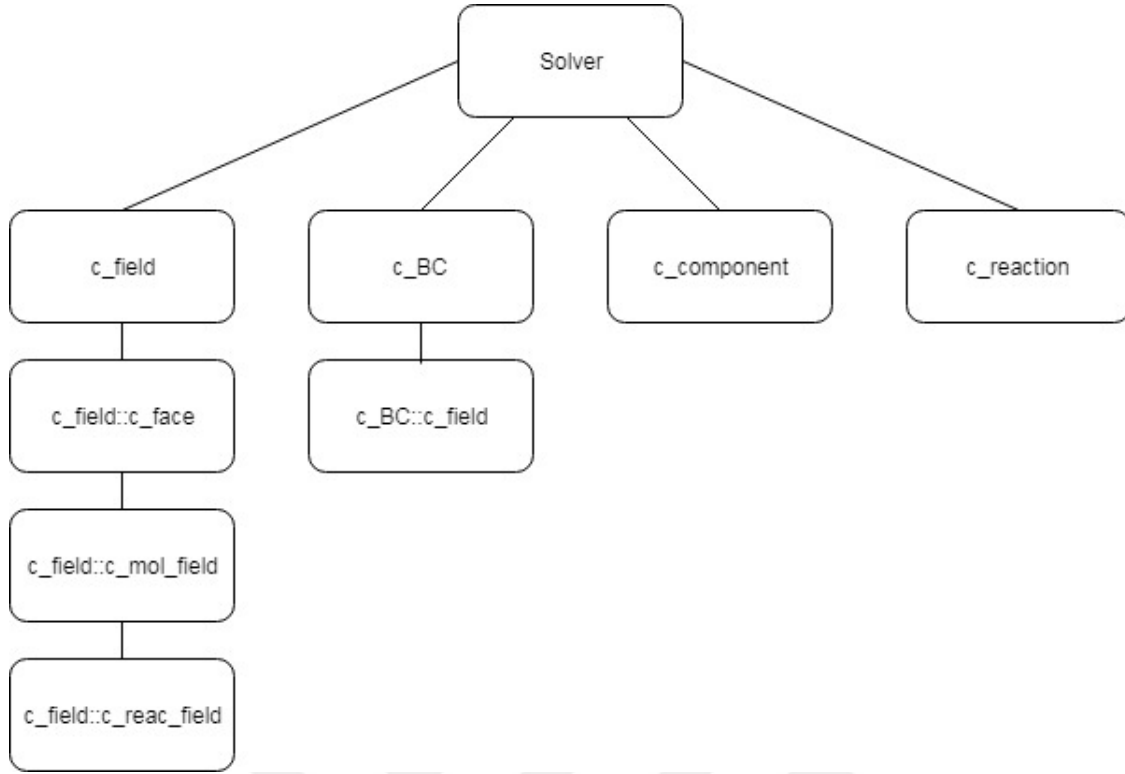
### 3. ÇÖZÜCÜNÜN GELİŞTİRİLMESİNDE KULLANILAN YÖNTEMLER

Bu tez dahilinde geliştirilen çözücü c++ dilinde yazılmıştır. CPU donanımı üzerindeki paralellik OpenMP kütüphanesi ile sağlanmıştır. GPU donanımı üzerindeki paralellik ise CUDA kütüphanesi ile sağlanmıştır. Tezin bu bölümünde çözücünün yazılmasında kullanılan yazılım geliştirme tekniklerinden bahsedilmektedir. Çözücü hem Windows hem Linux işletim sistemlerinde sorunsuz olarak çalışabilmektedir. Çözücü test edililen Microsoft Windows işletim sisteminde Microsoft derleyicisi ve CUDA Developer Kit 9.0 derleyicisi ile derlenmiştir. Çözücü Linux sisteminde de GCC derleyicisi ve CUDA Development Kit 9.0 derleyicisi ile derlenmiştir.

#### 3.1 Nesne Tabanlı Programlama

Bu tez dahilinde geliştirilen çözücü C++ dilinde yazılmıştır. C++ dili nesne tabanlı bir dildir ve çözücüde C++ dilinin nesne tabanlılığından faydalanılmıştır [12]. Çözücüde kullanılan nesne sınıflarının şeması Şekil 3.1'de gösterilmektedir.

Çözücüde "c\_field" adı verilen nesne çözülmekte olan hücrelerin sayısı kadar oluşturulmaktadır. Bu nesnede çözüm yapılan hücrelerin geometrik değerleri, akışkanlar dinamiği değerleri ve sayısal yöntemlerin değerleri tutulmaktadır. "c\_field" nesnesinin altındaki "c\_field::c\_face" nesnesi çözüm yapılan hücrelerin yüzlerinin geometrik değerleri, akışkanlar dinamiği değerleri ve sayısal yöntemlerin değerleri tutulmaktadır. "c\_field" nesnesinin altındaki "c\_field::c\_mol\_field" nesnesi birden çok kimyasal türün olduğu durumda, çözüm yapılan hücrelerin içindeki kimyasal türlerin transport denklemi değişkenlerini tutmaktadır. "c\_field::c\_mol\_field" nesnesinin "c\_field" nesnesinden farklı olmasının nedeni, çözüm esnasında akış problemindeki toplam kimyasal tür tipi sayısının kullanıcıdan alınmıyor olması ve her problem için farklı sayıda nesne bulunması gerekliliğidir. "c\_field" nesnesinin altındaki "c\_field::c\_reac\_field" nesnesi kimyasal reaksiyonların olduğu durumda, çözüm yapılan hücrelerin içindeki kimyasal



Şekil 3.1: Çözücüdeki nesne sınıfları

reaksiyonların reaksiyon hızı gibi değişkenlerini tutmaktadır. "c\_field::c\_reac\_field" nesnesinin "c\_field" nesnesinden farklı olmasının nedeni, çözüm esnasında akış problemindeki toplam kimyasal reaksiyon sayısının kullanıcıdan alınmıyor olması ve her problem için farklı sayıda kimyasal reaksiyon nesnesi bulunması gerekliliğidir.

"c\_BC" nesnesi çözülen problemdeki toplam sınır koşulu sayısı kadar oluşturulmaktadır ve problemin sınır koşullarının değerlerini tutmaktadır. "c\_BC" nesnesinin altındaki "c\_BC::c\_field" nesnesinin amacı sınır koşullarının uygulanışında sınır koşulunun hayalet hücre olarak kabul edildiği durumlarda hayalet hücrenin akışkanlar dinamiği ve sayısal yöntemler değişkenlerini tutmaktır.

"c\_component" nesnesi çözülen problemdeki kimyasal türlerin termodinamik ve transport sabitlerini tutmaktadır. Çözülen problemdeki kimyasal tür sayısı kadar oluşturulur.

"c\_reaction" nesnesi çözülen problemdeki kimyasal reaksiyonların sabitlerini tutar. Çözülen problemdeki kimyasal reaksiyon sayısı kadar oluşturulur.

### 3.2 Girdi-Çıktı Dosyaları

Çözücünün kullanımında girdi olarak 7 adet dosya kullanılmaktadır. Bu dosyalardan ilki konfigürasyon dosyasıdır. Konfigürasyon dosyasının amacı kullanıcıdan kullanılacak sayısal metotları, bu metotlarda kullanılan değişkenleri, analizde kullanılacak kimyasal tür sayısı ve kimyasal reaksiyon sayıları gibi bilgileri almaktadır. Konfigürasyon dosyası ayrıca kullanıcıdan diğer dosyaların isimlerini almaktadır. Konfigürasyon dosyası aynı zamanda simülasyon kontrol parametrelerini de içermektedir.

Girdi olarak alınan diğer dosya çözüm ağı dosyasıdır. Çözücü çözüm ağlarını gambit .neu uzantılı ASCII dosyası olarak almaktadır. Çözücü üç farklı çözüm ağı hücresi tipi ile çalışabilmektedir. Bu hücre tipleri, dört yüzlü (tetrahedron), üçgen prizma (wedge) ve altı yüzlü (hexahedron) olarak belirlenmiştir. Bu hücrelerin hepsi üç boyutlu hücrelerdir. Çözücüde iki boyutlu analizler, üç boyutlu ve bir hücre kalınlığındaki çözüm ağları kullanılarak yapılabilmektedir. İki boyutlu analizlerde analiz yapılmadığı boyut yönündeki sınır koşulları boş sınır koşulu olarak belirlenir.

Girdi olarak alınan üçüncü dosya sınır koşulu dosyasıdır. Bu dosyada, sınır koşullarının tipleri ve sınır koşullarında kullanılacak parametreler kullanıcıdan alınmaktadır.

Girdi olarak alınan dördüncü dosya tipi kimyasal tür dosyasıdır. Bu dosyada, kullanıcıdan, analizde kullanılacak kimyasal türlerin isimleri, termodinamik özellikleri ve taşınım özellikleri gibi parametreler alınmaktadır.

Girdi olarak alınan beşinci dosya akışkan özellikleri dosyasıdır. Akışkanın ısı sığası ve oluşum entalpisi gibi özellikleri kullanıcıdan bu dosya ile alınmaktadır.

Girdi olarak alınan altıncı dosya, reaksiyonların bilgilerinin kullanıcıdan alındığı dosyadır. Bu dosya ile alınan değişkenlerden bazıları reaksiyonların stokiyometrik ve Arrhenius denklemi katsayılarıdır.

Çözücünün kullandığı yedinci girdi dosyası analiz yeniden başlatma dosyasıdır. Bu dosya, içerisinde çözüm ağındaki bütün hücrelerin temel değişkenlerini barındırır. Bu dosya, analiz önceden kalınan noktadan tekrar başlatılmak istendiğinde kullanılmaktadır. Bu dosyanın bir diğer amacı da herhangi bir nedenden ötürü analiz yarıda kaldığında çözümü yedeklemektir.

Çözücü çıktı olarak üç adet dosya oluşturmaktadır. Bu dosyalardan ilki çözüm dosyasıdır. Çözüm dosyası .vtk uzantılıdır ve VTK [26] protokolüne uygun ASCII dosya oluşturmaktadır. Çözüm dosyasında analizdeki bütün hücrelerin önemli olan değişkenleri mevcuttur. Çözüm dosyaları açık kaynak bir yazılım olan PARAVIEW [28] kullanılarak görselleştirilebilmektedir. Bu tez dahilindeki görselleştirmeler PARAVIEW [28] yazılımı kullanılarak oluşturulmuştur.

Çözücünün çıktı olarak verdiği ikinci dosya tipi yüzey dosyasıdır. Yüzey dosyası, kullanıcı tarafından verilen bir sınır koşulundaki bütün yüzeylerin bilgilerini içermektedir. Çözücü tarafından oluşturulan üçüncü dosya ise yukarıda bahsedilen yeniden başlatma dosyasıdır. Oluşturulan bu dosyaların yazılma sıklığı kullanıcı tarafından belirlenmektedir.

### **3.3 Paralel Programlama Metodları**

Yüksek başarılı hesaplamaya açısından bakıldığında çözücünün bir iterasyonu iki bölüme ayrılabilir. Bu bölümler çözülecek matrisin hazırlandığı kısım ve hazırlanan matrisin çözüldüğü kısımdır. Çözücüde, matris çözme kısmının iterasyon süresinin yaklaşık %40 'ı olduğu görülmüştür. Bu bilgi ışığında GPU donanımının sadece matris çözüm kısmında kullanılması kararı verilmiştir. Bu kararın bir nedeni de literatürde, GPU donanımının matris çözme operasyonlarındaki performansının yüksek olduğundan bahsedilmesidir.

Çözücüde, matrislerin hazırlandığı kısımlar OpenMP [18] kütüphaneleri kullanılarak döngü temelli bir metod ile paralelleştirilmiştir. Bu metod, bütün çekirdeklerin ortak kullandığı bir hafızadan okunan değişkenler ile çalışan döngüyü hesaplama çekirdeklerine dengeli bir şekilde dağıtarak çalışmaktadır. Bu metodu, alternatifi olan ve çözüm ağındaki hücreleri ayrıştırmaya dayanan metottan daha verimsizdir fakat çözüm ağı ayrıştırma işleminden kaçınmak için seçilmiştir. Ayrıca bu paralelleştirme metodu sadece ortak hafızaya sahip donanımlar üzerinde çalışabildiği için sonsuza kadar ölçeklendirilememektedir fakat yeni teknoloji sunucu donanımlarının birden çok işlemci ile çalışabilmesinden dolayı kabul edilebilir ölçeklere ulaşabilmektedir.

Matris çözücü kısmı paralelleştirmesi için kullanıcı iki seçenekten birini seçebilmek-

tedir. Bu seçeneklerden ilki, matris çözüme işleminden önce yapılan AMG ön işleminin ve iteratif matris çözücüsünün ikisinde OpenMP kütüphanesi yardımı ile CPU donanımı üzerinde çalıştığı seçenektir. Bu seçeneklerden ikincisi ise AMG ön işleminin OpenMP ile CPU donanımı üzerinde yapıldığı sonra matris değişkenlerinin bilgisayarın ana hafızasından GPU hafızasına gönderildiği ve iteratif matris çözücünün GPU donanımında yapıldığı seçenektir. AMG ön işleminin CPU donanımında yapılmasının nedeni AMG operasyonlarının doğası gereği paralel performansının düşük olup CPU donanımında daha hızlı çözülmesidir.





## 4. DOĞRULAMA ÇALIŞMALARI

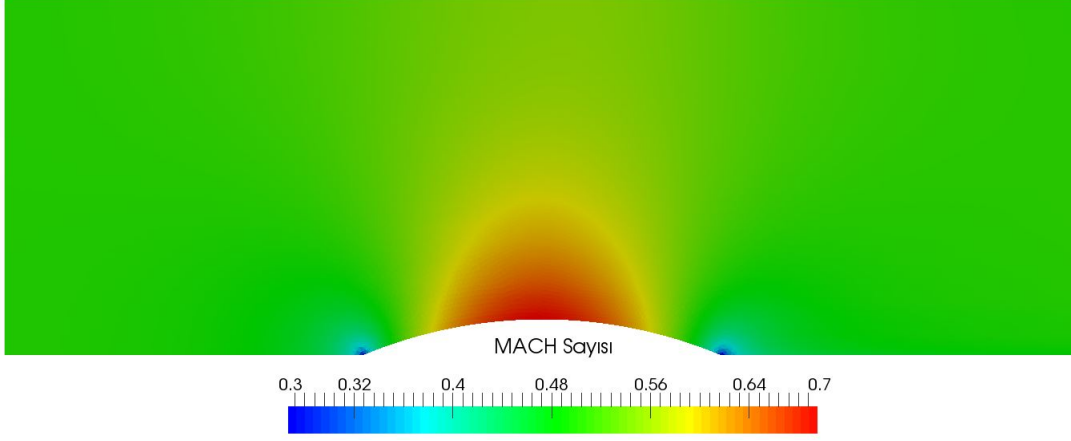
Tez kapsamındaki çözücüde yapılan doğrulama çalışmaları bu kısımda raporlanmıştır. Çözücünün transport denklemlerindeki konveksiyon terimlerini çözüm yönteminin ve çözücünün basınç-hız arasındaki bağıntıyı basınç merkezli çözüm algoritması ile çözme doğruluğu, tümsek üzerindeki iki boyutlu akış problemi ile test edilmiştir. Çözücünün difüzyon akıları hesaplamak için kullandığı sayısal şemaların ve HAD metodlarının doğruluğu, düz plaka üzerinde laminar akış problemi ile incelenmiştir. Zamanda integrasyonun başarımı Sod Shock Tube problemi ile doğrulanmıştır. Kimyasal tür taşınımı, kimyasal çözüm algoritmaları ve türbülans modellerinin doğrulanması Sandia Flame D çalışması ile doğrulanmıştır.

### 4.1 İki boyutlu tümsek üzerinde akış

#### 4.1.1 İki boyutlu tümsek üzerinde ses altı akış

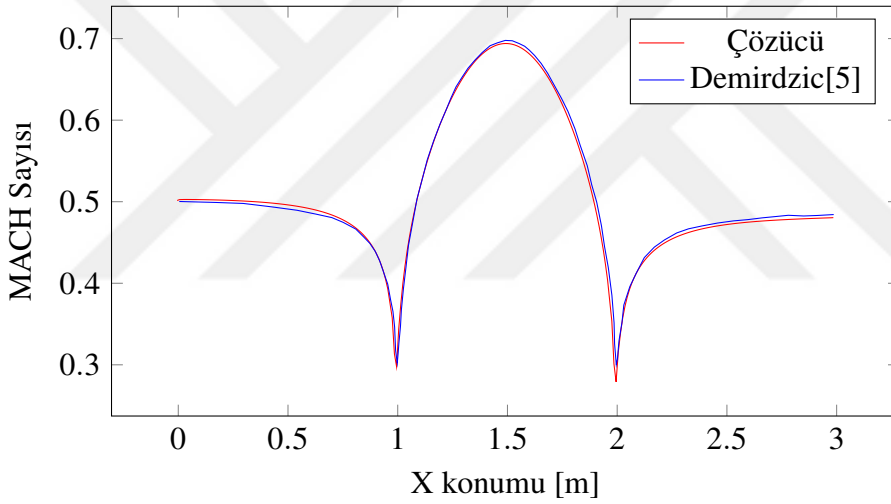
İki boyutlu tümsek üzerinde ses altı akış problemi Demirtzic'in [5] makalesinden alınmıştır. Akışkan olarak hava kullanılan tümsek probleminde Euler denklemleri çözülmüştür. Problem geometrisi ve kullanılan çözüm ağı Şekil 4.3'de gösterilmiştir. Çözüm için 400x100 çözünürlüğünde altıgen hücreli bir çözüm ağı kullanılmıştır.

Problem geometrisinin genişliği 3 metre ve yüksekliği 1 metredir. Geometrinin ortasındaki tümsek  $x = 1m$  ile  $x = 2m$  arasındadır. Tümseğin boyunun yüksekliğine oranı %10 olarak verilmiştir. Geometrinin sol tarafındaki sınır koşulu giriş sınır koşulu olarak belirlenmiştir ve giriş hızı 0.5 Mach olarak belirlenmiştir. Geometrinin sağındaki sınır koşulu basınç çıkışı olarak belirlenmiştir. Geometrinin altındaki ve üstündeki sınırlar duvar sınır koşulu olarak belirlenmiştir. Problemin Mach sayısı eş eğrileri Şekil 4.1'de gösterilmiştir.



Şekil 4.1: Tümsek üzerinde ses altı akış Mach sayısı eş eğrileri.

Problemin alt tarafındaki duvarın üzerindeki Mach sayılarının Demirtzic [5] ile karşılaştırılması Şekil 4.2'de görülmektedir.



Şekil 4.2: Tümsek üzerinde ses altı akış alt duvar Mach sayısı karşılaştırılması.

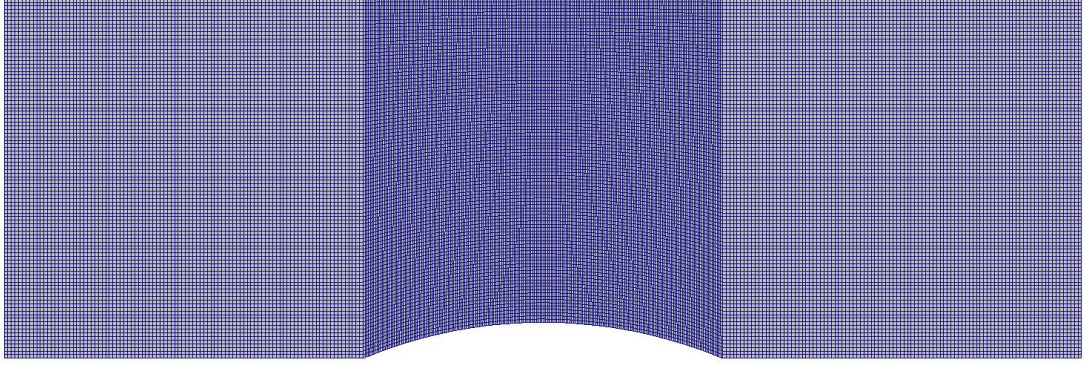
Şekilde görüldüğü üzere çözücünün sonucu, referans alınan çözüm ile çok yakın çıkmıştır. Bu sonuç ile çözücünün ses altı akışlardaki basınç-hız bağlantı algoritması doğrulanmıştır.

#### 4.1.2 İki boyutlu tümsek üzerinde transonik akış

İki boyutlu tümsek üzerinde transonik akış problemi, İki boyutlu tümsek üzerinde ses altı akış probleminde olduğu gibi Demirtzic'in [5] makalesinden alınmıştır. Problem

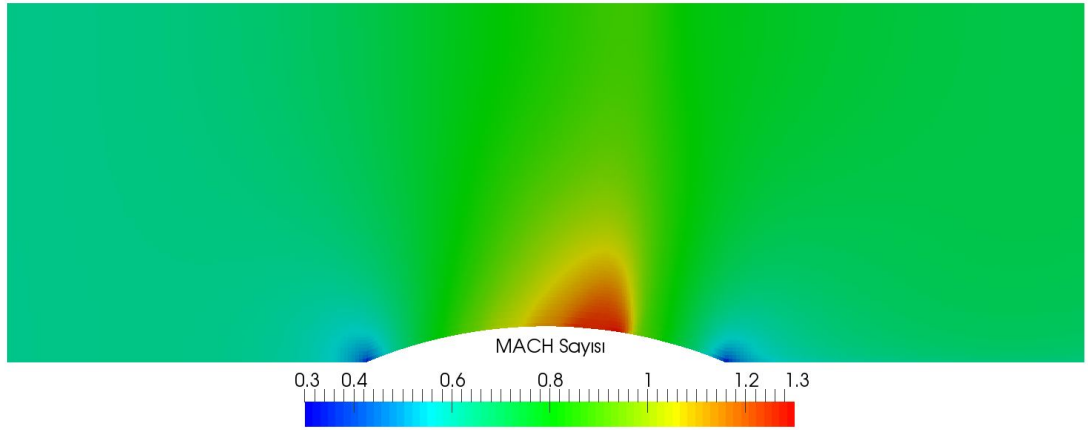


geometrisi ve problemde kullanılan çözüm ağı Şekil 4.3’de gösterilmiştir. Çözüm için 400x100 çözünürlüğünde altıgen hücreli bir çözüm ağı kullanılmıştır.



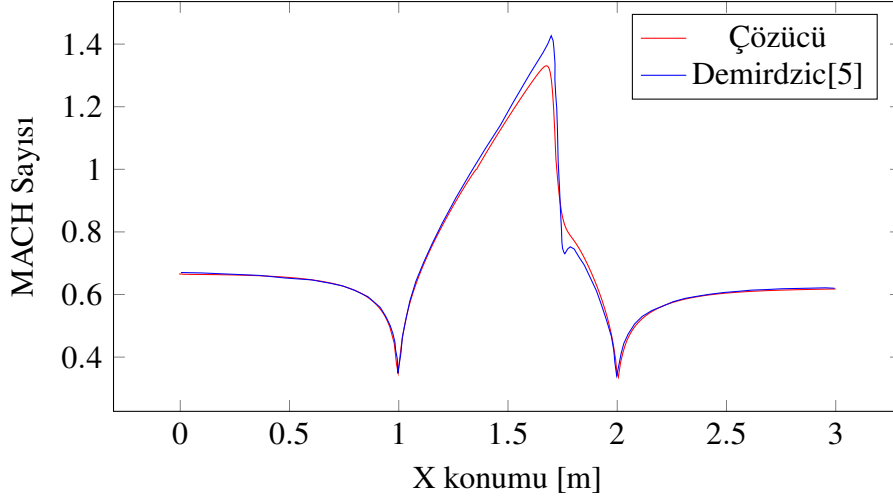
Şekil 4.3: Tümsek üzerinde ses altı ve transonik akış için çözüm ağı.

Problem geometrisinin genişliği 3 metre ve yüksekliği 1 metredir. Geometrinin ortasındaki tümsek  $x = 1m$  ile  $x = 2m$  arasındadır. Tümseğin boyunun yüksekliğine oranı %10 olarak verilmiştir. Geometrinin sol tarafındaki sınır koşulu giriş sınır koşulu olarak belirlenmiştir ve giriş hızı 0.675 Mach olarak belirlenmiştir. Geometrinin sağındaki sınır koşulu basınç çıkışı olarak belirlenmiştir. Geometrinin altındaki ve üstündeki sınırlar duvar sınır koşulu olarak belirlemiştir. Problemin Mach sayısı eş eğrileri Şekil 4.4’de gösterilmiştir.



Şekil 4.4: Tümsek üzerinde transonik akış Mach sayısı eş eğrileri.

Problemin alt tarafındaki duvarın üzerindeki mach sayılarının Demirtzic [5] ile karşılaştırılması Şekil 4.5’de görülmektedir.

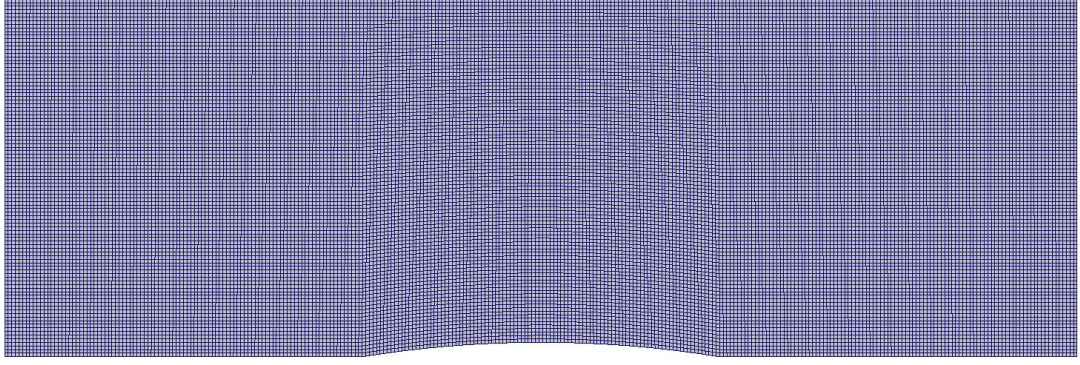


Şekil 4.5: Tümsek üzerinde transonik akış alt duvar Mach sayısı karşılaştırılması.

Şekilde görüldüğü üzere çözücünün sonucu ile referans sonuç birbirine kabul edilir yakınlıktadır. Çözücünün uzaydaki ayrıştırma şeması birinci mertebeden olduğundan dolayı referans çözümde görülen "dispersion" hataları çözücünün sonucunda görülmemiştir. Bu sonuç ile çözücünün transonik akışlardaki basınç-hız bağlantı algoritması doğrulanmıştır.

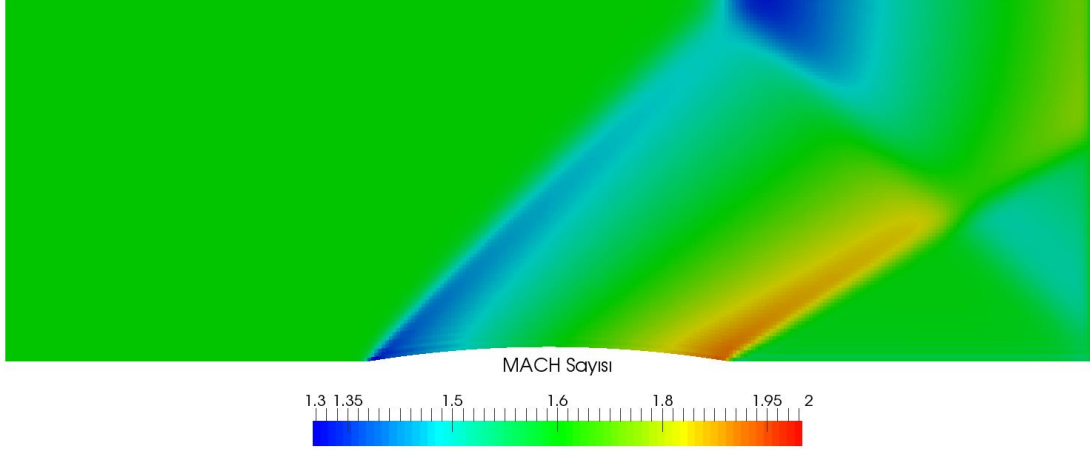
#### 4.1.3 İki boyutlu tümsek üzerinde ses üstü akış

İki boyutlu tümsek üzerinde ses üstü akış problemi Demirtzic'in [5] makalesinden alınmıştır. Problemin çözümünde Euler hava ile çözülmüştür. Problem geometrisi ve problemde kullanılan çözüm ağı Şekil 4.6'de gösterilmiştir. Çözüm için 300x100 çözünürlüğünde altıgen hücreli bir çözüm ağı kullanılmıştır.



Şekil 4.6: Tümsek üzerinde ses üstü akış çözüm ağı.

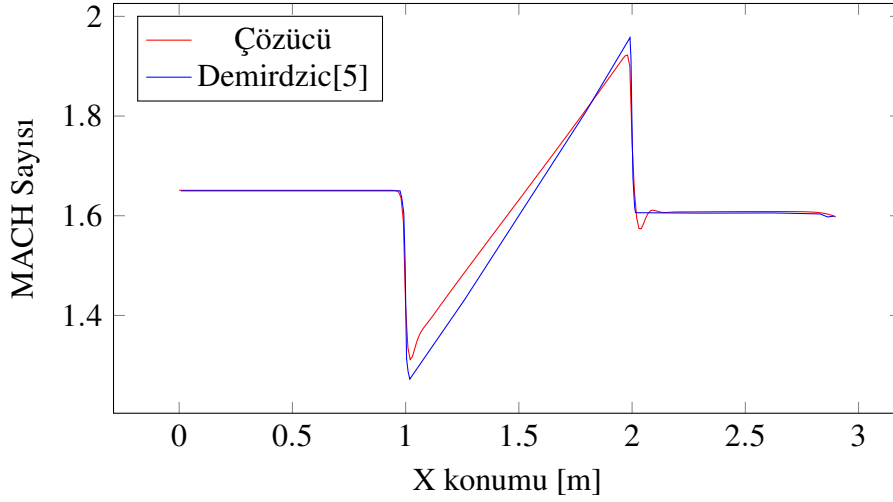
Problem geometrisinin genişliği 3 metre ve yüksekliği 1 metredir. Geometrinin ortasındaki tümsek  $x = 1m$  ile  $x = 2m$  arasındadır. Tümseğin boyunun yüksekliğine oranı %4 olarak verilmiştir. Geometrinin sol tarafındaki sınır koşulu giriş sınır koşulu olarak belirlenmiştir ve giriş hızı 1.65 Mach olarak belirlenmiştir. Geometrinin sağındaki sınır koşulu basınç çıkışı olarak belirlenmiştir. Geometrinin altındaki ve üstündeki sınırlar duvar sınır koşulu olarak belirlemiştir. Problemin Mach sayısı eş eğrileri Şekil 4.7’de gösterilmiştir.



Şekil 4.7: Tümsek üzerinde ses üstü akış Mach sayısı eş eğrileri.

Problemin alt tarafındaki duvarın üzerindeki mach sayılarının Demirtzic [5] ile karşılaştırılması Şekil 4.8’de görülmektedir.

Şekilde görüldüğü üzere çözücünün sonucu ile referans sonuç birbirine kabul edilir yakınlıktadır. Çözücünün uzaydaki ayrıştırma şeması birinci mertebeden olduğundan



Şekil 4.8: Tümsek üzerinde ses üstü akış alt duvar Mach sayısı karşılaştırılması.

dolayı referans çözümde görülen "dispersion" hataları, transonik çalışmada olduğu gibi burada da görülmemiştir. Bu sonuç ile çözücünün ses üstü akışlardaki basınç-hız bağlantı algoritması doğrulanmıştır.

#### 4.2 Düz plaka üzerinde laminar akış

Düz plaka üzerinde laminar akış problemi, literatürdeki HAD çözücülerinin doğrulanmasında sık olarak kullanılan bir problemdir. Düz plaka üzerinde laminar akış problemi çözücünün difüzyon terimi çözümünü doğrulama amacıyla çalışılmıştır. Düz plaka üzerinde laminar akış problemi 1 feet uzunluğundaki plaka üzerinde oluşan sınır tabakayı araştırmak amacı ile çözülmektedir. Problemin geometrisi 1 feet uzunluğunda kaymayan duvar sınır koşuluna sahiptir. Bu sınır koşulundan önce ise kayan duvar sınır koşulu vardır. Geometrinin sol tarafı giriş sınır koşulu sağ tarafı ise çıkış sınır koşuludur. Geometrinin üzerindeki sınır koşulu ise yine çıkış sınır koşuludur. Problemin girişindeki toplam sıcaklık 300 K olarak verilmiştir. Yine problemin girişindeki toplam basınç  $1000000 \text{ N/m}^2$  olarak girilmiştir. Problemin çıkışında ise  $97250 \text{ N/m}^2$  statik basınca sahip çıkış sınır koşulu yer almaktadır. Bu sınır koşulları problemde 0.2 MACH sayısına sahip bir akış oluşturmaktadır. Probleme başlangıç koşulu olarak bu hız ve çıkış statik basıncı verilmiştir. Bu hızın 1 feet plaka ile birimsizleştirilen Rey-

nolds sayısı 1301233.166 olarak hesaplanmaktadır. Normalde bu yükseklikteki Reynolds sayılarında plaka üzerindeki akışlar türbülanslı olabilmektedir fakat bu problemdeki akış laminar olarak kalmaktadır.

Çözüm ağı oluşturulurken kaymayan duvar koşulu olan sınırın en yakınındaki hücrenin bölgesel Reynolds sayısının 1 olmasına dikkat edilmiştir. Problemin çözümünde kullanılan çözüm ağı Şekil 4.10 de görülmektedir.

Problemin çözümü olan MACH sayısı eş eğrileri ölçekli olarak Şekil 4.9 de gösterilmiştir.

Çözücünden alınan sonuçlar aşağıda verilen Blasius [20] denklemlerinin çözümünden elde edilen sonuç ile karşılaştırılmıştır.

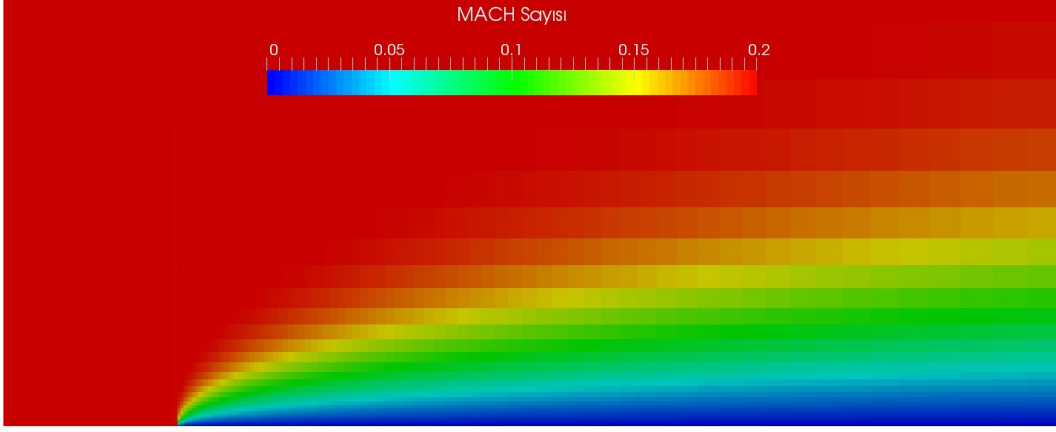
$$\begin{aligned} 2f''' + ff'' &= 0 \\ \frac{y}{\delta} = 0 : f\left(\frac{y}{\delta}\right) &= 0, f'\left(\frac{y}{\delta}\right) = 0 \\ \frac{y}{\delta} \rightarrow \infty : f'\left(\frac{y}{\delta}\right) &= 1 \end{aligned} \quad (4.1)$$

Denklemler çözüldükten sonra yüzey sürtünme katsayısı aşağıdaki hali alır.

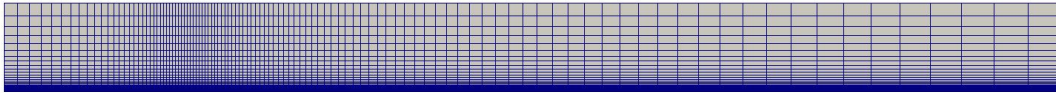
$$cf \approx \frac{0.664}{\sqrt{Re_x}} \quad (4.2)$$

Problemin çözümünün analitik sonuçlarla karşılaştırması Şekil 4.11 de gösterilmektedir.

Laminar düz plaka üzerindeki akış problemi çözümünde. Çözücünün sonucunun, Blasius analitik çözümüne çok yakın çıktığı görülmektedir. Bu çalışma ile çözücünün difüzyon terimi ayrıştırma yöntemleri doğrulanmıştır.

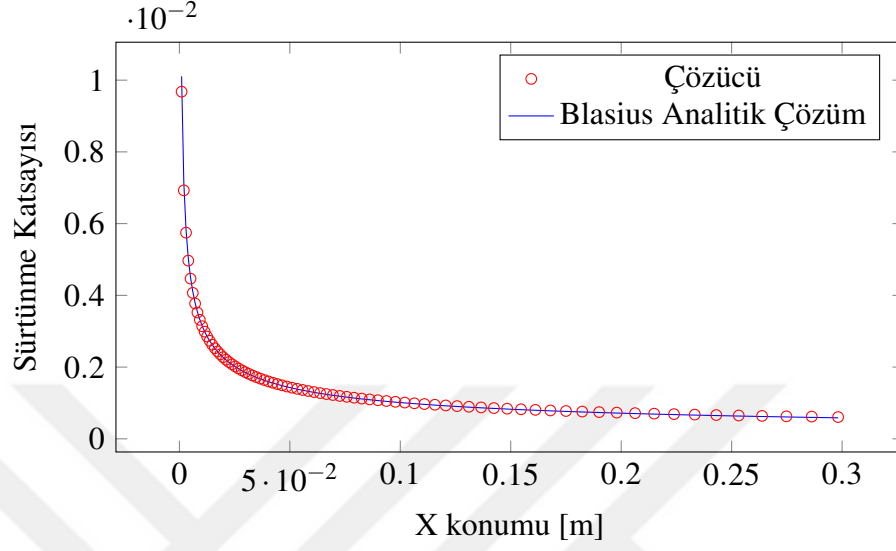


Şekil 4.9: Düz plaka üzerinde laminar akış Mach sayısı eş eğrileri ölçekli şekli.

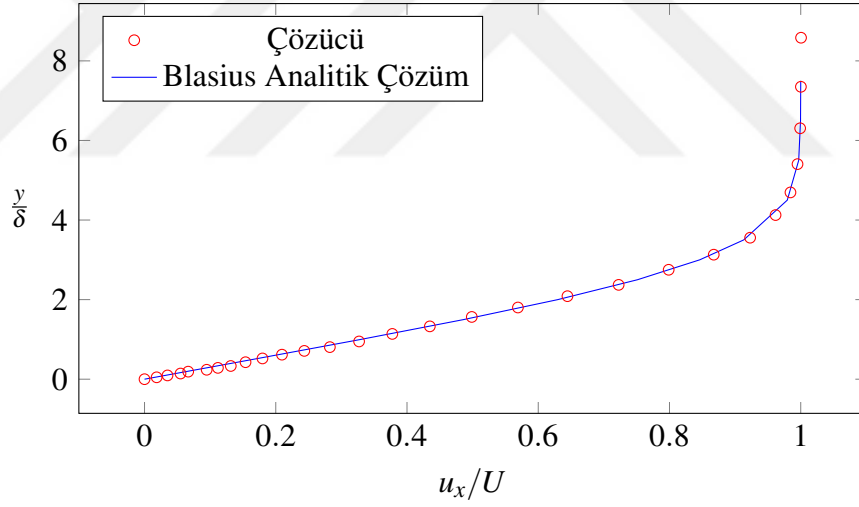


Şekil 4.10: Düz plaka üzerinde laminar akış çözüm ağı.





[a]



[b]

Şekil 4.11: Düz plaka üzerinde laminar akış sürtünme katsayısı (a) ve hız profili (b) karşılaştırması.

### 4.3 Sod Shock Tube doğrulama çalışması

Sod Shock tube problemi [21] , zamanda integrasyonun başarımını doğrulama amacı ile çözülmüştür. Sod Shock tube problemi her tarafı duvar olan iki boyutlu 1 ft x 0.1 ft boyutlara sahip dikdörtgen bir geometriye sahiptir. Problemdaki bütün sınır koşulları kayan duvar sınır koşulu olarak belirlenmiştir. Problemin iki yarısı ilk anda bir zar ile ayrılmaktadır. Analizin başlangıç koşulu problemin sağ ve solu olarak ikiye ayrılmıştır. Problemin başlangıç koşulları aşağıdaki tabloda verilmiştir.

Çizelge 4.1: Sod Shock Tube probleminin başlangıç koşulları.

Bölge	Basınç [Pa]	Sıcaklık [K]
Sol	68947.5	288.89
Sağ	6894.75	231.1

Sod Shock Tube problemi  $t = 2.11725E-04$  saniye anına kadar çözülmüştür. Bu andaki basınç eş eğrileri Şekil 4.12’de gösterilmiştir.

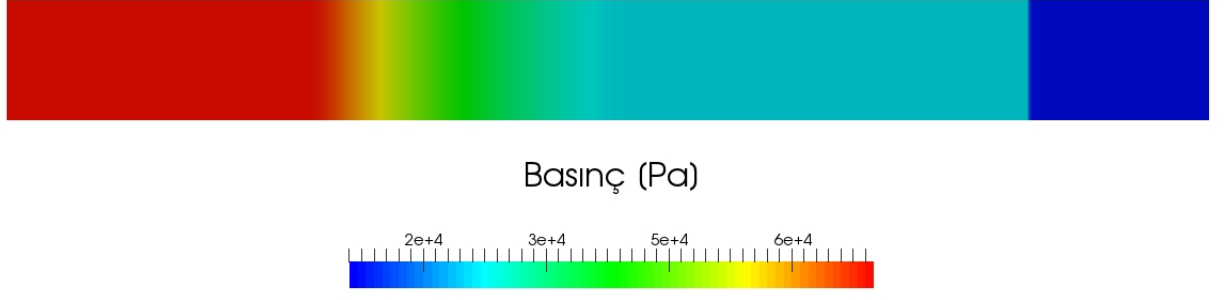
Problemin analitik sonucu aşağıdaki denklemlerin çözümü ile çıkmaktadır.

$$\frac{P_4}{P_1} = \frac{P_2}{P_1} \left[ 1 - \frac{\frac{a_1}{a_4}(\gamma_4 - 1)\left(\frac{P_2}{P_1} - 1\right)}{\sqrt{2\gamma_1 \left[ 2\gamma_1 + (\gamma_1 + 1)\left(\frac{P_2}{P_1} - 1\right) \right]}} \right]^{-\frac{2\gamma_4}{\gamma_4 - 1}} \quad (4.3)$$

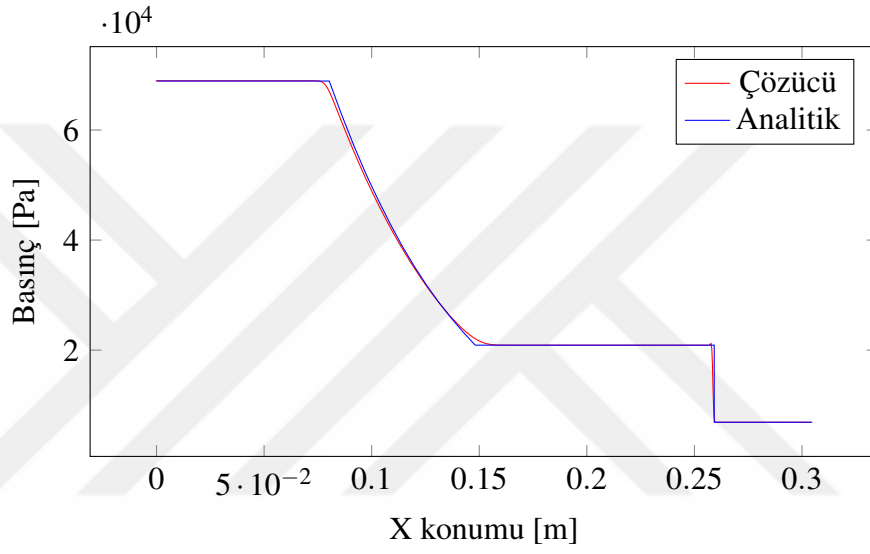
Bu denklemde 1 indisi ile gösterilen yer tüpün sağ tarafında kalan kısmı temsil etmektedir. Bu denklemde 4 indisi ile gösterilen yer tüpün sol tarafında kalan kısmı temsil etmektedir. 2 indisi ile gösterilen yer ise şok dalgaları geçtikten sonra tüpün orta kısmında oluşan yeri temsil etmektedir.

Çözücünün sonuçları ile analitik sonuçların karşılaştırması Şekil 4.13’de ve Şekil 4.14’de görülmektedir.



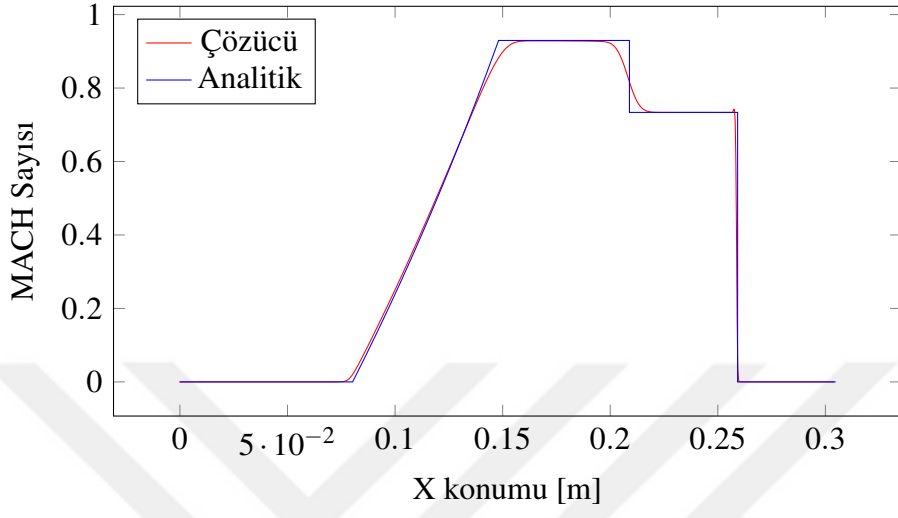


Şekil 4.12: Sod Shock tube problemi Basınç eş eğrileri.

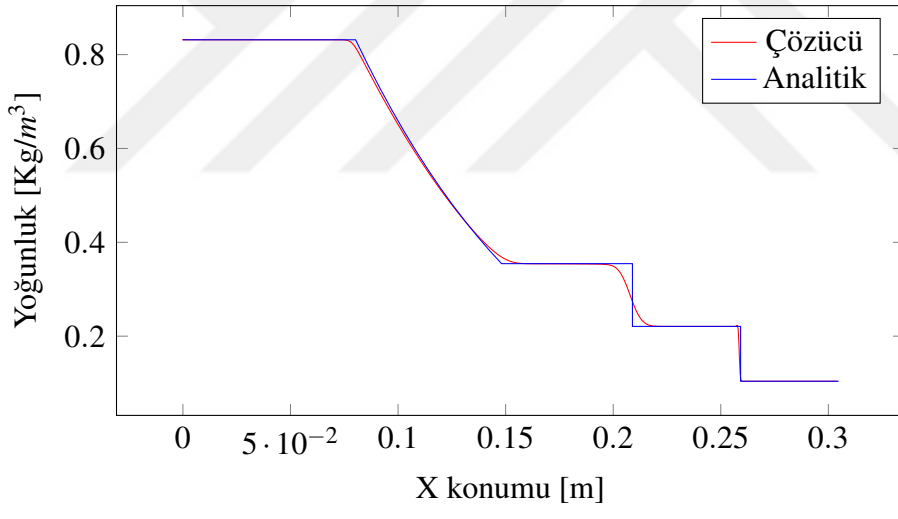


Şekil 4.13: Sod shock tube problemi basınç karşılaştırması.

Şekilde görüldüğü üzere çözücünün sonucu ile bahsedilen analitik sonuçlar birbirine çok yakın çıkmıştır. Bu çalışma ile çözücünün zamana bağlı ayrıştırma yöntemleri doğrulanmıştır.



[a]



[b]

Şekil 4.14: Sod shock tube problemi MACH sayısı (a) ve yoğunluk (b) karşılaştırması.

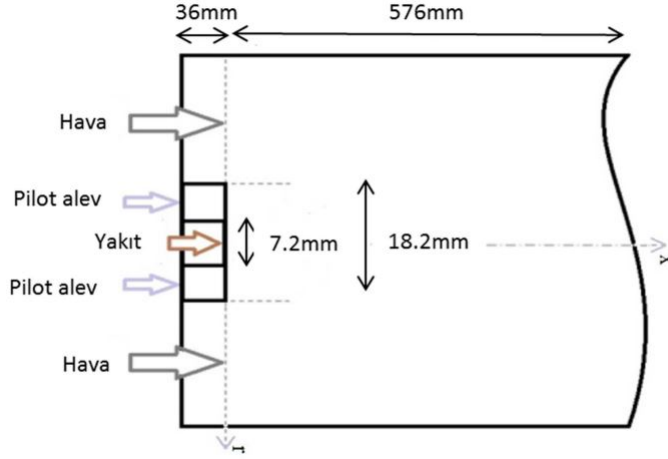
#### 4.4 Sandia Flame D doğrulama çalışması

Sandia Flame D [22] doğrulama çalışması; kimyasal tür taşınımı, kimyasal reaksiyon ve türbülans modellerinin doğrulanması amacıyla çalışılmıştır. Bu doğrulama çalışmasında yüksek sıcaklıktaki bir pilot alevi, önceden karışmış %25  $CH_4$  ve %75 hava karışımını tutuşturmuştur. Pilot alev çıkışı ana yakıt-hava çıkışının etrafında silindirik olarak bulunmaktadır. Test düzeneğinin fotoğrafı Şekil 4.15’de görülmektedir. Test düzeneğinin şematiği Şekil 4.16’de gösterilmektedir



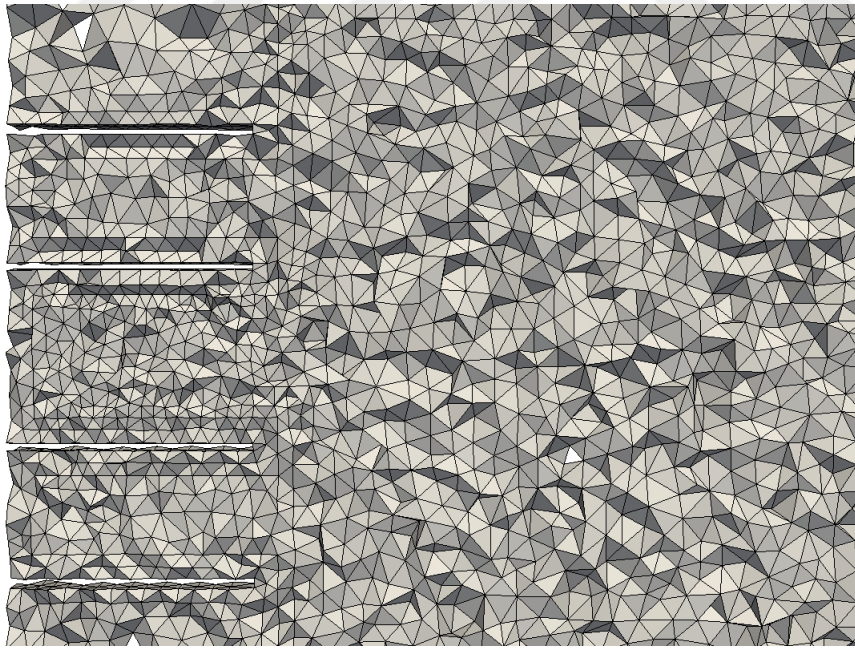
Şekil 4.15: Flame D test düzeneği fotoğrafı [22] .

Problem geometrisinde ana jetin çapı 7.2 mm olarak verilmiştir. Pilot jetinin iç çapı 7.7 mm , dış çapı 18.2 mm olarak verilmiştir, bu bilgiler ışığında ana jetin duvar kalınlığı 0.25 mm olarak hesaplanmaktadır. Yakıcının toplam çapı 18.9 mm olarak verilmiştir bu bilgi ışığında da pilot jetinin duvar kalınlığının 0.35 mm olduğu hesaplanmıştır. Yakıcıyı kapsayan rüzgar tünelinin çıkışı 30 cm x 30 cm olarak verilmiştir.



Şekil 4.16: Flame D test düzeneği şematiği [37] .

Bu problemin çözümünde 3 milyon tetradedral hücreli bir çözüm ağı kullanılmıştır. Çözümün basitliği açısından yakıcı duvarlarının akışa etkisinin olmadığı varsayıp problemdeki bütün duvar sınır koşulları kayan sınır koşulu olarak belirlenmiştir. Kullanılan çözüm ağının yakıcı çıkışına yakın olan tarafının ayrıntısı Şekil 4.17’de gösterilmiştir.



Şekil 4.17: Flame D problemi çözüm ağı ayrıntısı.

Sandia Laboratuvarları tarafından yapılmış Sandia piloted flame deneylerinde birçok

değişik ana karışım hızı ve pilot alev test edilmiştir. Bu tez kapsamında bu deneylerden D harfi ile adlandırılan deneyin koşullarının analizi yapılmıştır. Bu koşulda, ana yakıt hava karışımının giriş hızı 49.6 m/s olarak verilmiştir. Pilot alev hızı ise 11.04 m/s olarak verilmiştir. Deneyin yapıldığı rüzgar tünelinin hızı ise 0.9 m/s'dir. Pilot alevi analizde yüksek sıcaklıkta çözüm ağına giren yanma ürünleri olarak modellenmiştir. Pilot alevinin sıcaklığı 1880 K ve phi değeri 0.77 olarak analize girilmiştir.

Problemin çözümünde 6 kimyasal tür ve 2 reaksiyon içeren global bir reaksiyon mekanizması kullanılmıştır. Kullanılan reaksiyon mekanizması Poinso'tun makalesinden [23] alınmıştır. Reaksiyon mekanizmasında kullanılan kimyasal türler sırasıyla  $O_2$ ,  $CH_4$ ,  $CO$ ,  $CO_2$ ,  $H_2O$  ve  $N_2$  olarak belirlenmiştir. Kimyasal reaksiyonlar ise aşağıdaki gibidir.



Kimyasal reaksiyonların Arrhenius katsayıları Tablo da verilmiştir.

Çizelge 4.2: Metan hava yanması global reaksiyon mekanizması [23].

	1. Reaksiyon	2. Reaksiyon
Aktivasyon Enerjisi	$3.55 \times 10^4$	$1.2 \times 10^4$
Sıcaklık Faktörü	0.0	0.8
İlk Reaktant Ön Faktör	0.5	1.0
İkinci Reaktant Ön Faktör	0.65	0.5

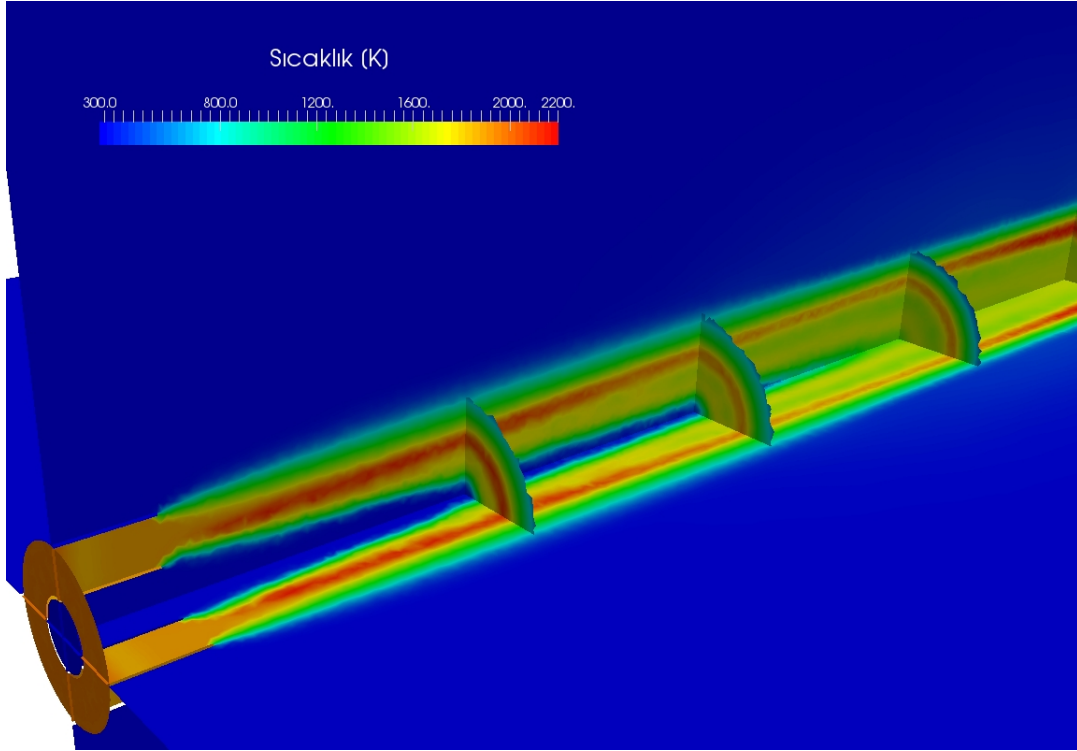
Analizin sıcaklık eş eğrileri Şekil 4.18'de gösterilmiştir. Analizin içinde çözülen 6 kimyasal türün kütlelesel derişim eş eğrileri ise Şekil 4.19'de gösterilmiştir.

Sandia tarafından yapılan testlerde yakıcının çıkışından itibaren 7.2 mm 14.4 mm 21.6 mm ve 108 mm uzaklıklarından radyal yönde sıcaklık ölçümleri yapılmıştır. Bu sıcaklık ölçümlerinin çözücü sonuçlarıyla karşılaştırılması Şekil 4.20 ve 4.21'de gösterilmektedir.

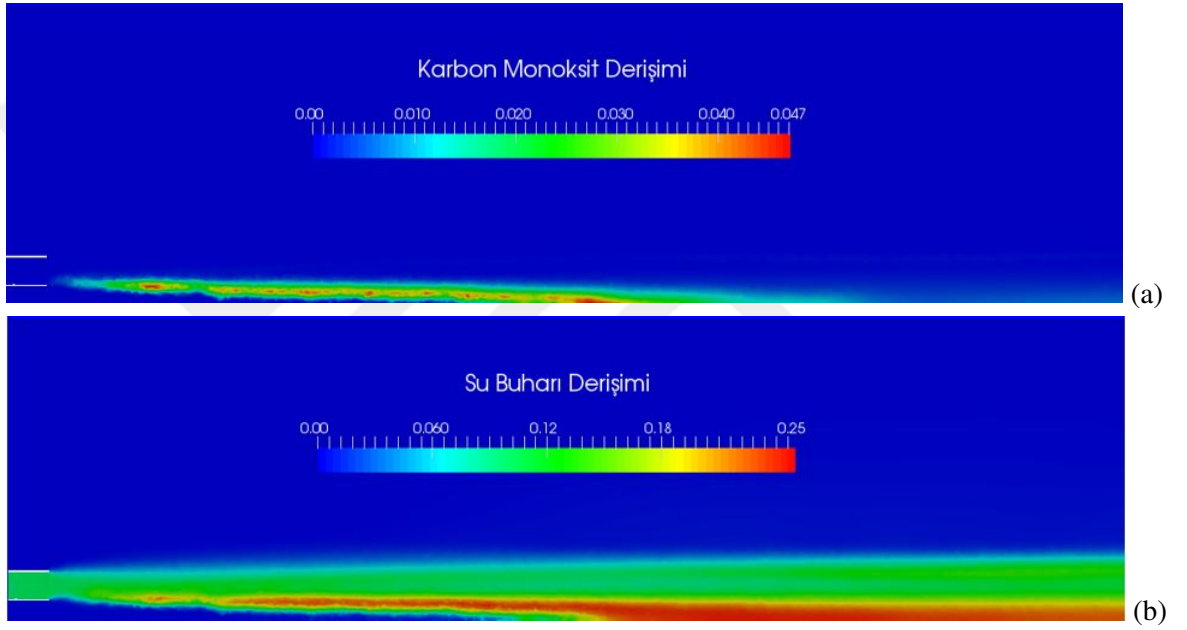
Çözücü sonuçları ile sıcaklık ölçümleri karşılaştırıldığında yakıcı geometrisinin hemen

önünden alınan analiz sonuçlarının sıcaklık ölçümlerinden uzak oluđu gözükmemektedir. Bu farkın nedeninin, kimyasal tür transport denkleminin difüzyon kısmında kullanılan kütle transferi katsayısının doğrudan kullanıcıdan alınmıyor olması olarak düşünölmektedir. Difüzyon katsayısı olması gereken değerdan az hesaplandığı için yakıcı çıkışında pilot alevi yeterince hızlı yayılamamıştır.

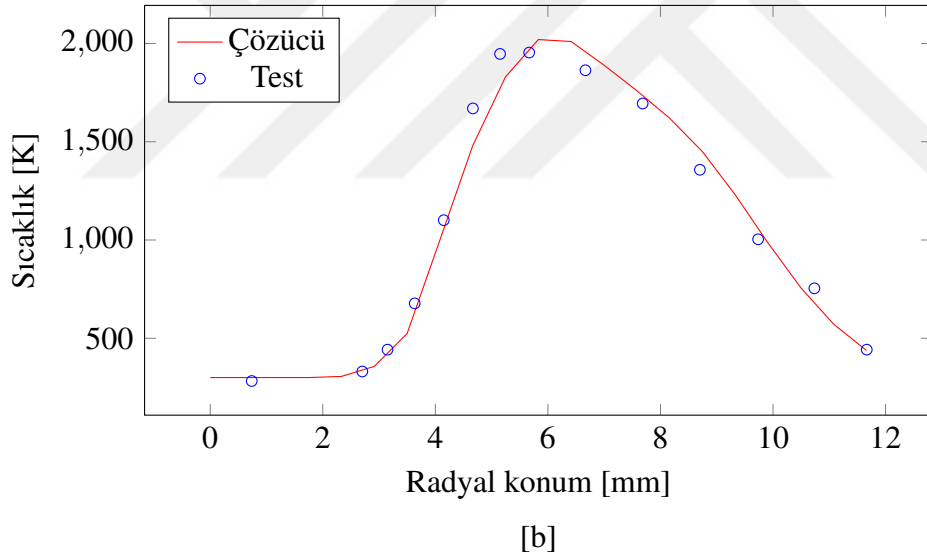
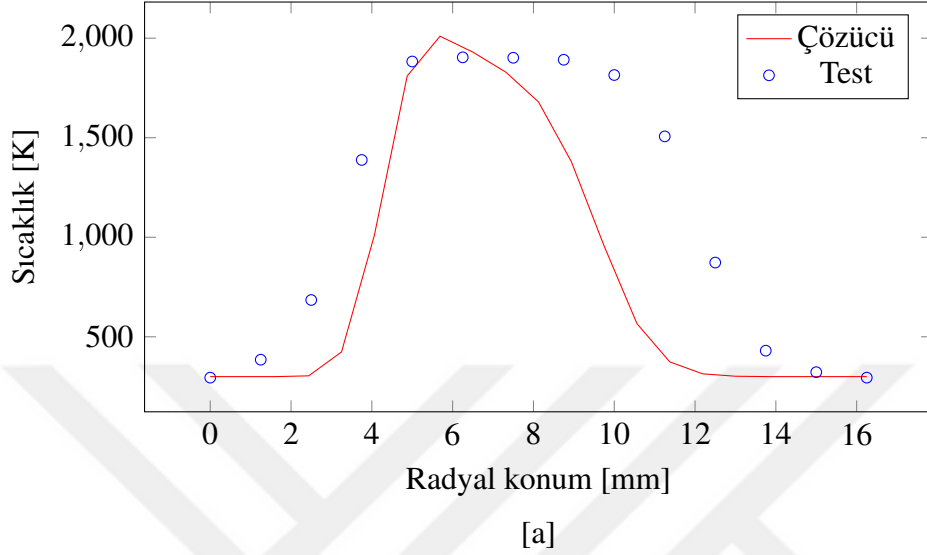
Yakıcı geometrisinin uzağından alınan çözücü sonuçlarının sıcaklık ölçümlerinden farklı çıkmasının sonucu, o bölgedeki türbölanslı viskozitenin fazla hesaplanmasından dolayı hızların düşmesi bu nedenle daha yüksek sıcaklık oluşması olarak düşünölmektedir.



Şekil 4.18: Flame D problemi sıcaklık eş eğrileri.

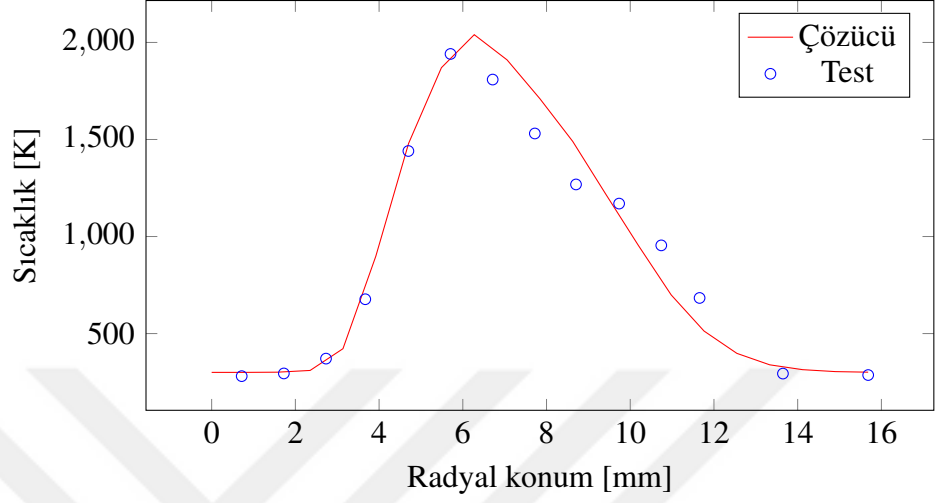


Şekil 4.19: Flame D problemi  $CO$  (a) ve  $H_2O$  (b) Kütlesel derişim eş eğrileri.

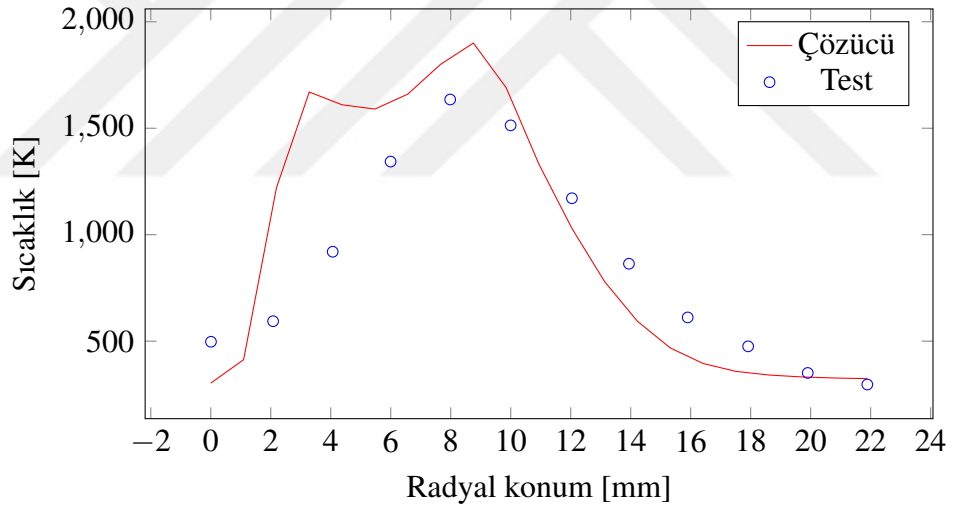


Şekil 4.20: Sandia Flame D  $x = 7.2$  mm (a) ,  $x = 14.4$  mm (b) konumlarında sıcaklık karşılaştırılması.





[a]



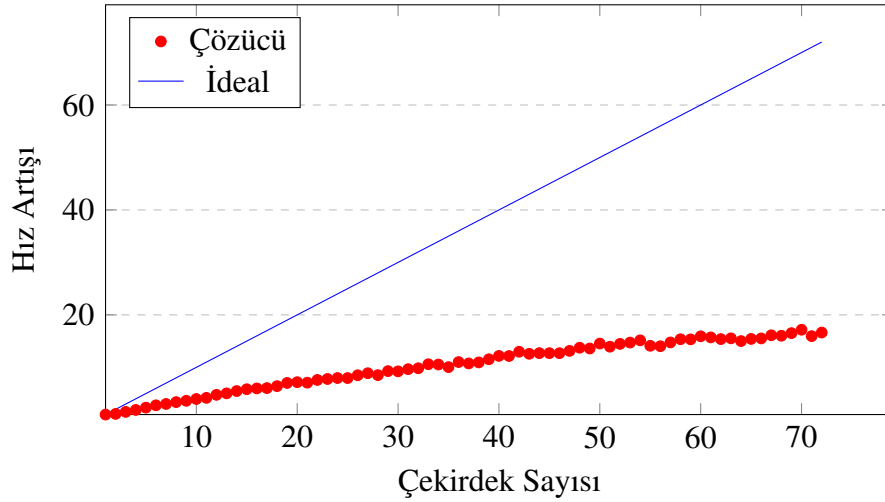
[b]

Şekil 4.21: Sandia Flame D  $x = 21.6$  mm (a) ,  $x = 108$  mm (b) konumlarında sıcaklık karşılaştırılması.



## 5. PERFORMANS ÖLÇÜMÜ ÇALIŞMALARI

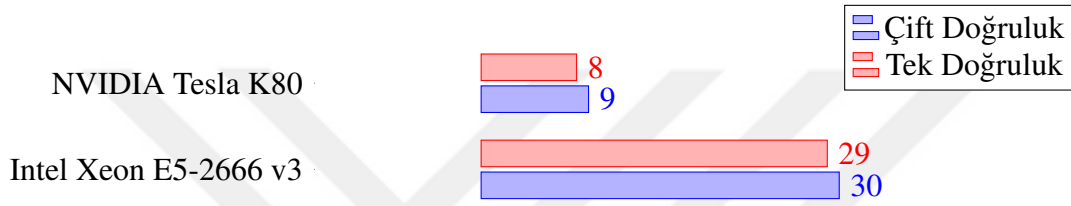
Bu bölümde çözücünün yüksek başarımlı hesaplama performansı incelenmiştir. Doğrulama için yapılan analizlerin iki boyutlu olanlarının hepsi kişisel dizüstü bilgisayar ile kabul edilebilir sürelerde çözülebilmektedir. Üç boyutlu, türbülanslı, yanmalı ve birden çok kimyasal türe sahip olan Sandia Flame D test çalışması ise Amazon tarafından sağlanan bulut yüksek başarımlı hesaplama hizmeti AWS EC2 ile çözülmüştür [24]. Performans kestirimi çalışmaları da yine bu hizmet yardımı ile yapılmıştır. CPU paralellik performansının kestirimi iki adet, 3 GHz saat hızına sahip Intel Xeon Platinum 8124M işlemcisi ile yapılmıştır. Bu iki işlemcinin oluşturduğu donanımda 36 adet fiziksel çekirdek bulunmaktadır. Her fiziksel çekirdeğin Hyper-Threading [25] teknolojisi ile kullandığı iki adet sanal çekirdek bulunmaktadır. Bu donanım ile yapılan performans çalışmasının sonucu Şekil 5.1’de gösterilmektedir. Bu grafikte çözücünün, matris çözümü haricinde yaptığı bütün işlemlerin toplam süresinin çekirdek sayısına göre hız artışı gösterilmiştir.



Şekil 5.1: CPU paralel durumda matris çözümü için harcanan zaman hariç hız artışı değerleri.

Çözücünün içerisinde kullanılan üçüncü parti matris çözme yazılımı olan ViennaCL,

OpenMP kütüphanesini kullanarak CPU donanımında paralel çalışabilmenin yanında CUDA kütüphanesini kullanarak GPU donanımında da paralel olarak çalışabilmektedir. GPU donanımı performans çalışmasında, Flame D problemi ilk olarak CPU paralel matris çözücü modunda sonrasında ise GPU paralel matris çözücü modunda çözülmüştür. Bu çalışma, çift doğruluğa sahip kayan nokta sayısı ve tek doğruluğa sahip kayan nokta sayısı için ayrı olarak tekrarlanmıştır. Bu çalışmada kullanılan GPU donanımı NVIDIA firmasının Tesla K80 olarak belirlenmiştir. CPU donanımı ise INTEL firmasının Xeon E5-2666 v3 donanımı olarak belirlenmiştir. Bu çalışmanın sonuçları Şekil 5.2’de gösterilmektedir.



Şekil 5.2: CPU ve GPU donanımında matris çözüm süreleri [saniye].

Bu grafikteki sonuçların karşılaştırmasının adil olabilmesi için GPU donanımındaki matris çözüm süreleri hesabında matris verilerinin GPU donanımına aktarılma süresi de katılmıştır. Yine performans ölçümünün gerçek hayat HAD analizlerini temsil edebilmesi açısından matris çözücü iterasyon sayısı 5 ile limitlenmiştir. Grafikte görüldüğü üzere, yaklaşık aynı fiyata sahip GPU donanımı aynı problemdeki matrisleri CPU donanımından üç buçuk kat hızlı çözmüştür.

## 6. SONUÇ VE ÖNERİLER

Tez kapsamında; üç boyutlu, basınç merkezli, kimyasal reaksiyon kabiliyetine sahip GPU destekli CPU paralel bir HAD çözücüsü geliştirilmiştir. Üçüncü parti matris çözücü yazılımı olan ViennaCL isimli kod çözücüye entegre edilmiştir.

Çözücü, Navier-Stokes ve Euler denklemlerini çözebilmektedir. Konveksiyon terimi için mertebeden "Upwind" ayrıştırma şemasına sahiptir. Difüzyon terimi için ise birinci mertebeden merkezi bir ayrıştırma şemasına sahiptir. Zaman terimi için ise birinci mertebe kapalı "Implicit" bir ayrıştırma şemasına sahiptir. Çözücü zamanda kapalı olduğu için doğrusal denklem çözücülerine ihtiyaç duymuştur. Çözücü dahilinde SOR doğrusal denklem çözücüsü bulunmaktadır. ViennaCL kütüphanesi sayesinde ise AMG yöntemi kullanan CG doğrusal denklem çözücüsü kullanılabilir. Çözücü, basınç merkezli bir basınç-hız birleştirme algoritması kullanmaktadır. Bu algoritma sıkıştırılabilir akışlar için uygun olup teorik olarak her MACH sayına sahip akışı çözebilmektedir fakat 2 MACH üzeri hızlardaki akışlar için doğrulanmamıştır.

Çözücü aynı zamanda RANS denklem seti ile türbülans modelleyebilmektedir. Türbülans modeli olarak  $k - \epsilon$  kullanılmaktadır.

Çözücüye kimyasal reaksiyonların çözümü için Arrhenius tipi FRC yanma modeli de eklenmiştir.

Çözücü C++ dilinde yazılmıştır. C++ dilinin nesne tabanlı olması özelliğini kullanmaktadır. Çözücü OpenMP kütüphanesi yardımı ile CPU donanımında paralel, CUDA kütüphanesi yardımı ile GPU donanımında paralel çözüm yapabilmektedir.

Viskoz akıların çözüm yöntemini doğrulamak için düz plaka üzerinde laminar akış problemi çalışılmıştır. Analiz sonuçları Blasius analitik çözümü ile karşılaştırılmıştır. Çözücü sonuçları, Blasius analitik çözümüne çok yakın çıkmıştır. Çözücünün konvektif akıları çözme yöntemini ve basınç temelli çözüm algoritmasını doğrulamak için dairesel tümseğe sahip kanal içerisinde ses altı ve ses üstü akış problemleri çözülmüştür.

Çözücünün sonuçları Demirdzic [5] tarafından yapılan analizler ile karşılaştırılmıştır. Ses altı olan doğrulama koşulunda çözücünün sonucu ile Dermirdzic tarafından bulunan sonuç arasında fark görülmemektedir. Transonik ve sesüstü olan akış sonuçlarının karşılaştırılmasında ise çözücünün Demirdzic'in sonuçlarına çok yakın olduğu fakat akış alanındaki keskin değişimleri Demirdzic kadar iyi öngöremediği görülmektedir. Bunun nedeni Demirdzic'in ikinci mertebeden ayrıştırma kullanıyor olması ve adaptif bir algoritma ile çözüm ağını iyileştiriyor olmasıdır. Beklendiği üzere keskin değişimleri yakalamakta, çözücünün birinci mertebeye şeması Demirdzic'in ikinci mertebeye şeması kadar başarılı olamamaktadır.

Çözücünün, zaman terimini ayrıştırma başarısının doğrulanması için Sod şok tüpü problemi çözülmüştür. Çözücünün sonuçları analitik sonuçlar ile karşılaştırılmıştır ve çözücünün, analitik sonuçları yakaladığı gözükmektedir.

Çözücünün, kimyasal reaksiyon çözme yeteneğinin, birden çok kimyasal türü çözme yeteneğinin ve jet karışımı alanındaki türbülans modelinin doğrulanması için üç boyutlu Sandia Flame D problemi çözülmüştür. Çözücünün sonuçları Sandia laboratuvarları tarafından yapılan sıcaklık ölçümleri ile karşılaştırılmıştır. Çözücü sonuçları ile test sonuçlarının orta alanda birbirine yakın olduğu görülmektedir. Çözücü sonuçları ile sıcaklık ölçümlerinin yakıcının hemen çıkışında birbirine çok yakın olmamasının nedeninin birinci mertebeye sayısal şema kullanımı olduğu kanaatine varılmıştır. Çözücü sonuçları ile sıcaklık ölçümlerinin yakıcıdan 108 mm uzaklıkta birbirine çok yakın çıkmamasının nedeninin türbülans modelinin türbülanslı viskoziteyi olması gerekenden biraz daha yüksek hesaplaması olduğu kanaatine varılmıştır.

Çözücünün performans ölçümü çalışmasında, yeterince iyi bir GPU donanımı ile çözücü içerisindeki iteratif seyrek matris çözücüsünün, aynı fiyatlı GPU donanımında, CPU donanımından çift veya tek doğruluktaki kayan nokta sayısı olmasından bağımsız olarak veri aktarım süresi dahil üç buçuk kat daha hızlı olduğu görülmektedir.

Çözücünün geliştirilmesi esnasında çok değerli bilgiler ve deneyimler elde edilmiştir. Geliştirilen çözücünün gelecekteki çalışmalara temel olacağı düşünülmektedir. Çözücünün geliştirilmesine bu tez dahilinde yapılan işler bittikten sonra da devam edilecektir.

## Kaynaklar

- [1] H. K. Versteeg and W. Malalasekera. An Introduction to Computational Fluid Dynamics The Finite Volume Method Second Edition Pearson 2007.
- [2] J. H. Ferziger and M. Peric. Computational Methods For Fluid Dynamics Springer 1997.
- [3] P. A. Davidson. Turbulence, an introduction for scientists and engineers Oxford 2004.
- [4] S. R. Turns. An Introduction to Combustion Concepts and Applications Third Edition McGraw Hill 2012.
- [5] I. Demirdzic and Z. Lilek and M. Peric. A collocated finite volume method for predicting flows at all speeds. In *International Journal for Numerical Methods in Fluids* 1993.
- [6] S. V. Patankar and D. B. Spalding. A CALCULATION PROCEDURE FOR HEAT MASS AND MOMENTUM TRANSFER IN THREE DIMENTIONAL PARABOLIC FLOWS. In *International Journal of Heat and Mass Transfer* 1972.
- [7] R. I. Issa. Solution of the Implicitly Discretised Fluid Flow Equations by Operator-Splitting. In *JOURNAL OF COMPUTATIONAL PHYSICS* 62, 4065 1985.
- [8] F. B. Hildebrand. Introduction to Numerical Analysis Second edition. Dover Publications 1974.
- [9] Jones, W. P., and Launder, B. E. The Prediction of Laminarization with a Two-Equation Model of Turbulence. In *International Journal of Heat and Mass Transfer*, vol. 15 1972.

- [10] B. J. McBride, S. Gordon and M. A. Reno Coefficients for Calculating Thermodynamic and Transport Properties of Individual Species. NASA Technical Memorandum 4513 1993.
- [11] W. Sutherland. The viscosity of gases and molecular force. *Philosophical Magazine Series 5* 1893.
- [12] A. Alexandrescu. Modern C++ Design: Generic Programming and Design Patterns Applied 1st Edition Addison-Wesley 2001
- [13] K. Rupp, F. Rudolf, J. Weinbub. ViennaCL - A High Level Linear Algebra Library for GPUs and Multi-Core CPUs. *Conference: Proceedings of the International Workshop on GPUs and Scientific Applications (GPUScA 2010)*.
- [14] M. R. Hestenes, E. Stiefel Methods of Conjugate Gradients for Solving Linear Systems *Journal of Research of the National Bureau of Standards Vol. 49, No.6, Research Paper 2379* 1952.
- [15] U. Trottenberg, C. W. Oosterlee, A. Schuller Multigrid Elsevier 2001
- [16] M. Wagner, K. Rupp, J. Weinbub. A Comparison of Algebraic Multigrid Preconditioners using Graphics Processing Units and Multi-Core Central Processing Units. *HPC '12 Proceedings of the 2012 Symposium on High Performance Computing* 2012.
- [17] P. Vanek. Algebraic Multigrid By Smoothed Aggregation For Second And Fourth Order Elliptic Problems. *Computing, vol. 56, no. 3, p. 179-196* 1996.
- [18] R. Chandra, R. Menon, L. D. David, K. D. Maydan, J. McDonald. Parallel Programming in OpenMP 1st Edition.
- [19] NVIDIA Corp. Cuda Toolkit Documentation v9.1.85 2018.
- [20] White, F.M. Viscous Fluid Flow , McGraw Hill Inc., New York 1974.



- [21] G. A. Sod. A Survey of Several Finite Difference Methods for Systems of Non-linear Hyperbolic Conservation Laws. *J. Comput. Phys.* 27 1978.
- [22] R. Barlow and J. Frank Piloted CH<sub>4</sub>/Air Flames C,D,E and F - Release 2.1. *Sandia National Laboratories* 2007.
- [23] B. Franzelli, E. Riber , L.Y.M. Giquel and T. Poinso Large-Eddy Simulation of combustion instabilities in a lean partially premixed swirled flame.
- [24] Amazon AWS EC2. <https://aws.amazon.com/tr/ec2/> 13.2.2018 .
- [25] D.T. Marr, F. Binns, D.L. Hill, G. Hinton, D.A. Koufaty, J. A. Miller, M. Upton Hyper-Threading Technology Architecture and Microarchitecture.
- [26] Kitware INC VTK User's Guide
- [27] F. Palacios, M. R. Colonno, A. C. Aranake, A. Campos, S. R. Copeland, T. D. Economon, A. K. Lonkar, T. W. Lukaczyk, T. W. R. Taylor, J. J. Alonso Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design. *AIAA Paper 2013-0287, 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Grapevine, Texas* 2013.
- [28] Ayachit, Utkarsh, The ParaView Guide: A Parallel Visualization Application, Kitware. 2015, ISBN 978-1930934306
- [29] F. Palacios, T. D. Economon, A. C. Aranake, S. R. Copeland, A. K. Lonkar, T. W. Lukaczyk, D. E. Manosalvas, K. R. Naik, A. S. Padron, B. Tracey, A. Variyar, J. J. Alonso Stanford University Unstructured (SU2): Open-source analysis and design technology for turbulent flows. *AIAA Paper 2014-0243, AIAA Science and Technology Forum and Exposition 2014: 52nd Aerospace Sciences Meeting, National Harbor, MD* 2014.
- [30] SU2 Code. <https://github.com/su2code/SU2> 3.10.2017 .

- [31] P. S. Pacheco. Parallel programming with MPI.
- [32] D. LaSalle, G. Karypis. Multi-Threaded Graph Partitioning. *27th IEEE International Parallel & Distributed Processing Symposium* 2013.
- [33] A. Lani An Object Oriented and High Performance Platform for Aerothermodynamics Simulation. *PhD. Thesis von Karman Institute for Fluid Dynamics* 2009.
- [34] E. A. Luke, X. L. Tong, J. Wu, L. Tang, P. Cinnella A Chemically Reacting Flow Solver for Generalized Grids. *AIAA International Chemical Engineering* 2003.
- [35] EIGEN <http://eigen.tuxfamily.org> 21.3.2017 .
- [36] ViennaCL <http://viennacl.sourceforge.net/> 18.4.2017 .
- [37] H. B. Bolat Yondes ve ters akisli hava parcalamali yakit-hava puskurtuculu bir yanma odasinin karakteristiklerinin hesaplamali akiskanlar dinamigi ile belirlenmesi *Yukse lisans tezi TOBB ETU* 2015.
- [38] Khronos Group OpenCL. <https://www.khronos.org/opencl/> 15.10.2017 .

## ÖZGEÇMİŞ

**Ad-Soyad** : Bertan ÖZKAN  
**Uyruğu** : Türkiye Cumhuriyeti  
**Doğum Tarihi ve Yeri** : 17.04.1992 Konya  
**E-posta** : bertanozkan@etu.edu.tr , bertan.ozkan@roketsan.com.tr  
**E-posta** : ozkanbertan@gmail.com

### ÖĞRENİM DURUMU:

- **Lisans** : 2015, TOBB ETÜ, Mühendislik Fakültesi, Makine Mühendisliği
- **Lisans** : 2015, TOBB ETÜ, Mühendislik Fakültesi, Bilgisayar Mühendisliği (yandal)

### MESLEKİ DENEYİM VE ÖDÜLLER:

Yıl	Yer	Görev
2016-	ROKETSAN A.Ş.	Sistem Tasarım Mühendisi (Aerotermodinamik Sorumlusu)

**YABANCI DİL:** İngilizce, Rusça

### TEZDEN TÜRETİLEN YAYINLAR, SUNUMLAR VE PATENTLER:

- Özkan, B., Uslu, S., 2018. Development of a Pressure Based, Unstructured, GPU Accelerated CFD Solver for Compressible Reacting Flows at all MACH Numbers, 10th International Conference on Computational Fluid Dynamics, July 9-13, Barcelona, Spain.

## **DİĞER YAYINLAR, SUNUMLAR VE PATENTLER:**

- Aksu, T., Kıyıcı, F., Kocaman, O. C., Uslu, S., 2015. A comparative aerodynamic study on cascade vanes for the cold stream type cascade reverser using Computational Fluid Dynamics (CFD), Ankara International Aerospace Conference AIAC-2015-010.

