

# OpenFOAM编程指南

---

OpenFOAM User Guide Chinese Edition

Christopher J. Greenshields 著

李东岳 译



Copyright©2011-2014 OpenFOAM Foundation.

This work is licensed under a **Creative Commons Attribution - NonCommercial NoDerivs 3.0 Unported License**.

版权声明©2011-2014 OpenFOAM Foundation

本指南遵守：**Creative Commons Attribution - NonCommercial NoDerivs 3.0 Unported License**.

## License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

### 1. Definitions

- a. **"Adaptation"** means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
- b. **"Collection"** means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the

selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

- c. **"Distribute"** means to make available to the public the original and copies of the Work through sale or other transfer of ownership.
- d. **"Licensor"** means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
- e. **"Original Author"** means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
- f. **"Work"** means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
- g. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
- h. **"Publicly Perform"** means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work

to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

- i. **"Reproduce"** means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

**2. Fair Dealing Rights.** Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

**3. License Grant.** Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections; and,
- b. to Distribute and Publicly Perform the Work including as incorporated in Collections.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats, but otherwise you have no rights to make Adaptations. Subject to 8(f), all rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights set forth in Section 4(d).

**4. Restrictions.** The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights

granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested.

- b. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.
- c. If You Distribute, or Publicly Perform the Work or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work. The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Collection, at a minimum such credit will appear, if a credit for all contributing authors of Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.
- d. For the avoidance of doubt:
  - i. **Non-waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
  - ii. **Waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this

License if Your exercise of such rights is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(b) and otherwise waives the right to collect royalties through any statutory or compulsory licensing scheme; and,

- iii. **Voluntary License Schemes.** The Licensor reserves the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License that is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(b).
- e. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation.

## **5. Representations, Warranties and Disclaimer**

UNLESS OTHERWISE MUTUALLY AGREED BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

**6. Limitation on Liability.** EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## **7. Termination**

- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Collections from You under this License, however, will not have their licenses

terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

- b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

## **8. Miscellaneous**

- a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
- b. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
- c. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
- d. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
- e. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

## Trademarks

ANSYS is a registered trademark of ANSYS Inc.

CFX is a registered trademark of Ansys Inc.

CHEMKIN is a registered trademark of Reaction Design Corporation

EnSight is a registered trademark of Computational Engineering International Ltd.

Fieldview is a registered trademark of Intelligent Light

Fluent is a registered trademark of Ansys Inc.

GAMBIT is a registered trademark of Ansys Inc.

Icem-CFD is a registered trademark of Ansys Inc.

I-DEAS is a registered trademark of Structural Dynamics Research Corporation

JAVA is a registered trademark of Sun Microsystems Inc.

Linux is a registered trademark of Linus Torvalds

OpenFOAM is a registered trademark of OpenCFD Ltd

ParaView is a registered trademark of Kitware

STAR-CD is a registered trademark of Computational Dynamics Ltd.

UNIX is a registered trademark of The Open Group

This *OpenFOAM Programmer's Guide, Chinese Edition* is translated into Chinese by Dongyue Li<sup>1</sup>(Chapter 1, 2), Xiangyu Gao<sup>2</sup>(Chaper 3).

<sup>1</sup>DYFLUID Studio, Torino, Piedmont, Italy, 10143. <sup>2</sup>Purdue University, West Lafayette, IN, USA, 47907

《*OpenFOAM 编程指南中文版*》由李东岳<sup>1</sup>翻译第一、二章，高翔宇<sup>2</sup>翻译第三章

<sup>1</sup>东岳流体，意大利皮埃蒙特区都灵市，10143. <sup>2</sup>普渡大学，美国印第安纳州西拉法叶，47907.



## 目录

<b>License</b> .....	2
<b>第一章</b> .....	11
1.1 坐标系统.....	11
1.2 张量.....	11
1.2.1 张量表示法.....	13
1.3 张量算术.....	13
1.3.1 内积.....	14
1.3.2 双内积.....	15
1.3.3 两个三阶张量的三内积.....	15
1.3.4 外积.....	15
1.3.5 矢量的叉乘.....	15
1.3.6 其他张量操作.....	16
1.3.7 几何变形和单位张量.....	16
1.3.8 张量等式.....	17
1.3.9 二阶张量算术.....	17
1.3.10 标量算术.....	18
1.4 OpenFOAM 张量类 .....	19
1.4.1 OpenFOAM 张量操作.....	19
1.5 量纲单位.....	21
<b>第二章</b> .....	23
2.1 微分算符.....	23
2.1.1 梯度.....	23
2.1.2 散度.....	24
2.1.3 旋度.....	24
2.1.4 拉普拉斯.....	24
2.1.5 时间导数.....	24
2.2 离散简述.....	25
2.2.1 OpenFOAM 链表和场.....	25
2.3 计算域离散.....	25
2.3.1 定义 OpenFOAM 网格.....	27
2.3.2 定义 geometricField.....	28
2.4 方程离散.....	29
2.4.1 拉普拉斯项.....	34
2.4.2 对流项.....	34
2.4.3 一阶时间导数.....	35
2.4.4 二阶时间导数项.....	35
2.4.5 散度项.....	35
2.4.6 梯度项.....	36
2.4.7 梯度-梯度的平方 .....	37
2.4.8 旋度.....	37
2.4.9 源项.....	37
2.4.10 其他显性离散.....	37

---

2.5 时间离散.....	38
2.5.1 OpenFOAM 中时间离散的代码.....	39
2.6 边界条件.....	39
2.6.1 物理边界条件.....	40
第三章 .....	41
3.1 圆柱绕流.....	41
3.1.1 问题阐述.....	42
3.1.2 potentialFoam 求解器的注意事项.....	43
3.1.3 网格的生成.....	43
3.1.4 边界条件和初始条件.....	45
3.1.5 运行算例.....	46
3.2 稳态后向台阶湍流模拟.....	49
3.2.1 问题阐述.....	49
3.2.2 网格的生成.....	50
3.2.3 边界条件和初始场.....	53
3.2.4 求解器的控制参数.....	53
3.2.5 运行算例和后处理.....	54
3.3 前向台阶的超音速绕流.....	54
3.3.1 问题阐述.....	54
3.3.2 网格生成.....	56
3.3.3 运行算例.....	57
3.3.4 练习.....	57
3.4 高压水箱放水问题.....	58
3.4.1 问题阐述.....	58
3.4.2 网格的生成.....	59
3.4.3 准备运行算例.....	61
3.4.4 运行算例.....	62
3.4.5 细化网格来获得更精确解.....	63
3.5 磁流体问题.....	63
3.5.1 问题阐述.....	64
3.5.2 网格的生成.....	65
3.5.3 运行算例.....	66

# 第一章

## 张量

这一章我们介绍张量和张量算术，以及他们在本手册中是如何进行表达的。然后我们解释张量和张量算术如何在 OpenFOAM 中的实现。

### 1.1 坐标系统

OpenFOAM 主要用来解决连续介质问题，例如：固体应力分析、液体气体流动问题以及物理材料的变形问题。因此，OpenFOAM 在三维空间和时间中处理用张量来进行描述的物理问题。参见图 1.1，OpenFOAM 使用的坐标系统符合右手规则。建立于右手规则的坐标系符合下述特征：三个坐标轴起源于  $O$  点，他们的名字为  $Ox$ 、 $Oy$ 、 $Oz$ ，当从  $Oz$  向上观测的时候（ $O$  点离观测者较近），从  $Ox$  上某点指向  $Oy$  上某点的圆弧为顺时针方向<sup>1</sup>。

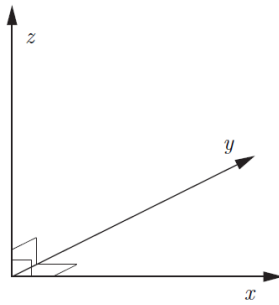


图 1.1 右手规则坐标轴

### 1.2 张量

张量用来描述从属某一空间的实体，其遵循一定的数学法则。简要说来，张量就是和一系列单位矢量相关的一系列的值；在 OpenFOAM 中，单位矢量  $\mathbf{i}_x$ ,  $\mathbf{i}_y$ ,  $\mathbf{i}_z$  和上文提及的符合右手规

---

<sup>1</sup> 本手册相对于 *OpenFOAM ProgrammersGuide* 对行距有所调整以保证页数匹配

则的坐标系相关。单位矢量是正交的，每个张量具有如下特征：

- **维度  $d$** ：我们用  $d$  来表示，在 OpenFOAM 中  $d = 3$ ；
- **阶（秩）  $r$** ：其为一个整数，表示分量的个数，例如我们可以这样表示： $d^r$ ；  
在 OpenFOAM 1.x 中，我们默认设置为 3 维（ $d = 3$ ），并提供 0 到 3 阶张量作为标准张量，对张量进行扩展也非常容易。我们熟知 0 阶张量和 1 阶张量代表标量和矢量，2 阶张量和 3 阶张量有些读者可能会感到有些陌生。为了文章的完整性我们把他们都进行了描述；
- **0 阶标量**：我们用 0 阶张量表示任何一个可以用单个数值表示的物理量，例如：质量  $m$ ，体积  $V$ ，压力  $p$ ，粘度  $\mu$ ；
- **1 阶矢量**：它用来表示具有大小和方向的物理量。他们通常的形式为： $\mathbf{a} = (a_1, a_2, a_3)$ ，其和笛卡尔坐标系的  $x, y, z$  相对应。有时我们用索引符  $i$  表示这个张量，例如  $a_i, i = 1, 2, 3$ ，但是本书中我们将其省略，因为我们处理的问题均为 3 维的；
- **2 阶张量**：我们用  $\mathbf{T}$  来表示， $\mathbf{T}$  具有 9 个元素，参见如下：

$$\mathbf{T} = T_{ij} = \begin{pmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{pmatrix} \quad (1.1)$$

由于其为 2 阶张量，因此分量  $T_{ij}$  使用两个索引符来表示，如上述，我们在下文省略  $ij$  索引符号。当  $i = j$  的时候， $T_{ij}$  为对角元素。那些  $i \neq j$  的元素为非对角元素。 $\mathbf{T}$  的转置矩阵如下：

$$\mathbf{T} = T_{ji} = \begin{pmatrix} T_{11} & T_{21} & T_{31} \\ T_{12} & T_{22} & T_{32} \\ T_{13} & T_{23} & T_{33} \end{pmatrix} \quad (1.2)$$

需要注意的是，我们通常将二阶张量称为张量，因为再往上的高阶张量非常罕见；

- **对称 2 阶张量**：对称张量指得是非对角元素关于对角元素是对称的。例如  $T_{ij} = T_{ji}$ ，在这种情况下，只有 6 个独立元素（ $T_{12} = T_{21}$ ， $T_{13} = T_{31}$ ， $T_{23} = T_{32}$ ）。在 OpenFOAM 中我们区分对称张量和非对称张量来节省内存。因为对称张量只需要存储 6 个元素，非对称张量需要存储 9 个元素。大多数连续介质问题中的张量为对称张量；
- **3 阶张量**：具有 27 个元素，我们用  $P_{ijk}$  来表示，由于其太复杂在这里我们不表示；
- **对称 3 阶张量**：OpenFOAM 将其定义为下述元素相等的 3 阶张量： $P_{ijk} = P_{ikj} = P_{jik} = P_{jki} = P_{kij} = P_{kji}$ ，因此它具有 10 个独立分量。更确切来说，3 个相同的矢量进行外积即可得到对称 3 阶张量，我们在 1.3.4 节中介绍外积；

### 1.2.1 张量表示法

本书是一本关于连续介质计算力学的书籍，主要处理 3 维空间和时间的复杂偏微分方程组。在开始后续章节之前我们很有必要了解一下张量的表示方法以防止在后续章节发生混乱。为了很好的理解方程组，我们希望读者接受这样一个概念：张量是一个整体而不是一系列标量的组合。另外，任何张量操作都是在这个整体上进行操作而不是对某个元素进行操作。

在本书下面的章节中，所有的张量（除了标量）我们都是用粗体，**a**。我们使用单一字符来表示张量，这更加的表明张量本身是一个实体，并且这样表示张量非常简洁。这种表达方式潜在的缺点就是张量的阶数并没有有效的表达。很明显，张量的阶数不能为 0。实际上，由于我们很清楚的了解各个张量代表的物理量的实际意义，缺少张量阶数的这种表达方式也不会对我们造成太大的麻烦。例如我们知道速度 **U** 的阶数为 1。

另外一个关于张量操作符的基本概念是张量和坐标系统的无关性。例如矢量 **a** 不管我们从哪个角度来观测它都是矢量 **a**。这意味着张量和坐标系统无关。然而其他符号，例如  $a_i$ ，它则和坐标系统有关，因为它表示的是张量的某个分量。这样的结果就是张量由一系列和坐标系统有关的元素值来表示，这些元素值依赖于坐标系统。

正如 1.2 节所说，对于张量的索引符表达方式将贯彻全书，它的主要目的是为了将张量运算拓展到其中的每个分量上。当使用索引符表示张量的时候，我们定义，如果在一项中某个索引符出现了两次，则这一项表示元素（例如第 1、2、3 元素）的相加。例如：

$$a_i b_i = \sum_{i=1}^3 a_i b_i = a_1 b_1 + a_2 b_2 + a_3 b_3 \quad (1.3)$$

利用简化的上述定义，在下面的章节中， $\sum$  将被省略。

## 1.3 张量算术

在这一章节我们描述 OpenFOAM 可用的张量算术。我们首先来看一下最常见的张量操作：加、减、乘、除。加法和减法只能同阶操作。他们的操作即为将张量对应的元素分别进行加减法。例如，矢量 **a** 和矢量 **b** 的减法即为：

$$\mathbf{a} - \mathbf{b} = a_i - b_i = (a_1 - b_1, a_2 - b_2, a_3 - b_3) \quad (1.4)$$

张量 **a** 和标量  $s$  的乘法也是可交换的，即为将张量的每个元素都乘以标量，例如：

$$\mathbf{sa} = sa_i = (sa_1, sa_2, sa_3) \quad (1.5)$$

在做张量 **a** 和标量  $s$  的除法的时候，标量  $s$  必须为分母，例如：

$$\mathbf{a}/s = a_i/s = (a_1/s, a_2/s, a_3/s) \quad (1.6)$$

依据上述基本定义，在接下来章节中，我们讨论一阶张量的更为复杂的乘积操作。

### 1.3.1 内积

张量的内积可以在阶数为 $r_1$ 的张量和阶数为 $r_2$ 的张量上进行操作。结果张量的阶数为 $r = r_1 + r_2 - 2$ ，下面我们描述至高 3 阶的张量内积：

- 矢量  $\mathbf{a}$  和矢量  $\mathbf{b}$  的内积是可交换的，并且结果为一个标量  $s$ ：

$$s = a_i b_i = a_1 b_1 + a_2 b_2 + a_3 b_3 \quad (1.7)$$

- 张量  $\mathbf{T}$  和矢量  $\mathbf{a}$  的内积结果为矢量  $\mathbf{b}$ ，我们用列向量来表示  $\mathbf{b} = \mathbf{T} \cdot \mathbf{a}$ （看起来更加方便）：

$$b_i = T_{ij} a_j = \begin{pmatrix} T_{11}a_1 + T_{12}a_2 + T_{13}a_3 \\ T_{21}a_1 + T_{22}a_2 + T_{23}a_3 \\ T_{31}a_1 + T_{32}a_2 + T_{33}a_3 \end{pmatrix} \quad (1.8)$$

如果  $\mathbf{T}$  是非对称张量，那么这个内积操作是非对称的，即  $\mathbf{b} = \mathbf{a} \cdot \mathbf{T} = \mathbf{T}^T \cdot \mathbf{a}$ ：

$$b_i = a_j T_{ji} = \begin{pmatrix} T_{11}a_1 + T_{21}a_2 + T_{31}a_3 \\ T_{12}a_1 + T_{22}a_2 + T_{32}a_3 \\ T_{13}a_1 + T_{23}a_2 + T_{33}a_3 \end{pmatrix} \quad (1.9)$$

- 两个张量  $\mathbf{T}$  和  $\mathbf{S}$  的内积也是一个张量，即  $\mathbf{P} = \mathbf{T} \cdot \mathbf{S}$ ， $\mathbf{P}$  元素的值这样计算：

$$P_{ij} = T_{ik} S_{kj} \quad (1.10)$$

并且这种运算是不可交换的，例如： $\mathbf{T} \cdot \mathbf{S} = (\mathbf{S}^T \cdot \mathbf{T}^T)^T$

- 矢量  $\mathbf{a}$  和三阶张量  $\mathbf{P}$  的内积结果为一个二阶张量  $\mathbf{T}$ ， $\mathbf{T} = \mathbf{a} \cdot \mathbf{P}$ ， $\mathbf{T}$  的元素为：

$$T_{ij} = a_k P_{kij} \quad (1.11)$$

同样，这个运算也是不可交换的，因此  $\mathbf{T} = \mathbf{P} \cdot \mathbf{a}$  的元素为：

$$T_{ij} = P_{ijk} a_k \quad (1.12)$$

- 二阶张量  $\mathbf{T}$  和三阶张量  $\mathbf{P}$  的内积为三阶张量  $\mathbf{Q}$ ， $\mathbf{Q} = \mathbf{T} \cdot \mathbf{P}$ ， $\mathbf{Q}$  的元素为：

$$Q_{ijk} = T_{il} P_{ljk} \quad (1.13)$$

同样，这个运算也是不可交换的，因此  $\mathbf{Q} = \mathbf{P} \cdot \mathbf{T}$  的元素为：

$$Q_{ijk} = P_{ijl} T_{lk} \quad (1.14)$$

### 1.3.2 双内积

两个二阶张量  $\mathbf{T}$  和  $\mathbf{S}$  的双内积结果为一个标量  $s$  ( $s = \mathbf{T} : \mathbf{S}$ ),  $s$  这样计算:

$$s = T_{ij}S_{ij} = T_{11}S_{11} + T_{12}S_{12} + T_{13}S_{13} + T_{21}S_{21} + T_{22}S_{22} + T_{23}S_{23} + T_{31}S_{31} + T_{32}S_{32} + T_{33}S_{33} \quad (1.15)$$

二阶张量  $\mathbf{T}$  和三阶张量  $\mathbf{P}$  的双内积结果为一个矢量  $\mathbf{a}$  ( $\mathbf{a} = \mathbf{T} : \mathbf{P}$ ):

$$a_i = T_{jk}P_{jki} \quad (1.16)$$

这个运算是不可交换的, 因此  $\mathbf{a} = \mathbf{P} : \mathbf{T}$  为:

$$a_i = P_{ijk}T_{jk} \quad (1.17)$$

### 1.3.3 两个三阶张量的三内积

两个三阶张量  $\mathbf{P}$  和  $\mathbf{Q}$  的三内积结果为一个标量  $s$  ( $s = \mathbf{P} : \mathbf{Q}$ ),  $s$  这样计算:

$$s = P_{ijk}Q_{ijk} \quad (1.18)$$

### 1.3.4 外积

张量以及矢量的外积这样定义:

- 两个矢量  $\mathbf{a}$  和  $\mathbf{b}$  的外积是不可交换的, 结果为张量  $\mathbf{T} = \mathbf{ab} = (\mathbf{ba})^T$ , 其这样计算:

$$T_{ij} = a_i b_j = \begin{pmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{pmatrix} \quad (1.19)$$

- 一个矢量  $\mathbf{a}$  和一个张量  $\mathbf{T}$  的外积为一个三阶张量  $\mathbf{P}$  ( $\mathbf{P} = \mathbf{aT}$ ), 其这样计算:

$$P_{ijk} = a_i T_{jk} \quad (1.20)$$

同样这个运算为不可交换的, 因此  $\mathbf{P} = \mathbf{T a}$  的结果为:

$$P_{ijk} = T_{jk} a_i \quad (1.21)$$

### 1.3.5 矢量的叉乘

叉乘只能用于矢量, 对于两个矢量  $\mathbf{a}$  和  $\mathbf{b}$ , 他们的叉乘为矢量  $\mathbf{c}$ , 其这样计算:

$$c = a \times b = e_{ijk} a_j b_k = (a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1) \quad (1.22)$$

其中 $e_{ijk}$ 这样定义：

$$e_{ijk} = \begin{cases} 0 & \text{当两个索引符相同的时候} \\ +1 & \text{ijk 索引符为关于 123 的偶次序} \\ -1 & \text{ijk 索引符为关于 123 的奇次序} \end{cases} \quad (1.23)$$

其中偶次序为偶数次交换相邻检索符得到 123, 231, 312, 奇次序为奇数次交换交换相邻检索符得到 132, 213, 321。

### 1.3.6 其他张量操作

下面我们描述 OpenFOAM 提供的一些不太常用的张量操作：

- **平方**：某个张量和自己本身的外积，例如对于矢量  $\mathbf{a}$  的平方为： $\mathbf{a}^2 = \mathbf{a}\mathbf{a}$ ；
- **n 次幂**：某个张量和自己本身的 n 次外积，例如对于矢量  $\mathbf{a}$  的 3 次幂为： $\mathbf{a}^3 = \mathbf{a}\mathbf{a}\mathbf{a}$ ；
- **模的平方**：某个张量和自己本身的阶次内积，例如对于 2 阶张量  $\mathbf{T}$  有 $|\mathbf{T}|^2 = \mathbf{T} : \mathbf{T}$ ；
- **模**：为模的平方的平方根，例如对于张量  $\mathbf{T}$ ， $|\mathbf{T}| = \sqrt{\mathbf{T} : \mathbf{T}}$ 。单位大小的矢量我们称为单位矢量；
- **最大元素**：张量的最大元素（包括符号），注意：非最大的模；
- **最小元素**：张量的最小元素；
- **平均元素**：张量元素的平均；
- **缩放**：正如名称所言，缩放用于使用某个张量的元素对另一个张量的元素进行缩放，这两个张量的阶数需要一致，例如：对  $\mathbf{b}$  矢量使用  $\mathbf{a}$  矢量进行缩放后为矢量  $\mathbf{c}$ ，计算如下：

$$c_i = \text{scale}(\mathbf{a}, \mathbf{b}) = (a_1 b_1, a_2 b_2, a_3 b_3) \quad (1.24)$$

### 1.3.7 几何变形和单位张量

2 阶张量可以定义为矢量的线性函数，例如：向量  $\mathbf{a}$  和向量  $\mathbf{b}$  可以通过  $\mathbf{T}$  联系起来 $\mathbf{a} = \mathbf{T} \cdot \mathbf{b}$ 。我们可以通过改变  $\mathbf{T}$  中的分量来使  $\mathbf{T}$  能够对张量实现不同的坐标变换，例如从  $x$ 、 $y$ 、 $z$  的坐标系统变换为  $x^*$ 、 $y^*$ 、 $z^*$  坐标系统。这种变换称之为张量转换。然而，标量在坐标系变换时不发生变化。矢量可实现下述变换：

$$\mathbf{a}^* = \mathbf{T} \cdot \mathbf{a} \quad (1.25)$$

二阶张量  $\mathbf{S}$  可以通过下述公式变换成为  $\mathbf{S}^*$ ：

$$\mathbf{S}^* = \mathbf{T} \cdot \mathbf{S} \cdot \mathbf{T}^T \quad (1.26)$$

单位张量定义为：它将某个张量变换为自己本身，即：



$$\mathbf{a} = \mathbf{I} \cdot \mathbf{a} \quad (1.27)$$

因此，单位张量为：

$$\mathbf{I} = \delta_{ij} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1.28)$$

其中 $\delta_{ij}$ 为 *Kronecker* 标示符。

### 1.3.8 张量等式

下面我们来列举一下常用的张量等式，我们假定相关的导数都是存在且连续的。其中  $s$  表示标量， $\mathbf{a}$  表示矢量：

$$\begin{aligned} \nabla \cdot (\nabla \times \mathbf{a}) &= 0 \\ \nabla \times (\nabla s) &= 0 \\ \nabla \cdot (s\mathbf{a}) &= s\nabla \cdot \mathbf{a} + \mathbf{a} \cdot \nabla s \\ \nabla \times (s\mathbf{a}) &= s\nabla \times \mathbf{a} + \mathbf{a} \times \nabla s \\ \nabla(\mathbf{a} \cdot \mathbf{b}) &= \mathbf{a} \times (\nabla \times \mathbf{b}) + \mathbf{b} \times (\nabla \times \mathbf{a}) + (\mathbf{a} \cdot \nabla)\mathbf{b} + (\mathbf{b} \cdot \nabla)\mathbf{a} \\ \nabla \cdot (\mathbf{a} \times \mathbf{b}) &= \mathbf{b} \cdot (\nabla \times \mathbf{a}) - \mathbf{a} \cdot (\nabla \times \mathbf{b}) \\ \nabla \times (\mathbf{a} \times \mathbf{b}) &= \mathbf{a}(\nabla \cdot \mathbf{b}) - \mathbf{b}(\nabla \cdot \mathbf{a}) + (\mathbf{b} \cdot \nabla)\mathbf{a} - (\mathbf{a} \cdot \nabla)\mathbf{b} \\ \nabla \times (\nabla \times \mathbf{a}) &= \nabla(\nabla \cdot \mathbf{a}) - \nabla^2 \mathbf{a} \\ (\nabla \times \mathbf{a}) \times \mathbf{a} &= \mathbf{a} \cdot (\nabla \mathbf{a}) - \nabla(\mathbf{a} \cdot \mathbf{a}) \end{aligned} \quad (1.29)$$

了解下面这个  $e - \delta$  等式有助于用户对张量在标识符下的操作有更好的理解：

$$e_{ijk}e_{irs} = \delta_{jr}\delta_{ks} - \delta_{js}\delta_{kr} \quad (1.30)$$

### 1.3.9 二阶张量算术

下面我们列举几个 2 阶张量的算法：

- **转置**：正如方程 2.1 所说， $\mathbf{T} = T_{ij}$ ， $\mathbf{T}^T = T_{ji}$ ；
- **symm、skew**：正如 1.2 节所说，我们有对称张量即其关于对角线是对称的： $\mathbf{T} = \mathbf{T}^T$ 。非对称张量 (skew 张量) 符合这种特性： $\mathbf{T} = -\mathbf{T}^T$ ，这意味着  $T_{11} = T_{22} = T_{33} = 0$ ，每个二阶张量都可以分为 symm 部分和 skew 部分即：

$$\mathbf{T} = 0.5 \underbrace{(\mathbf{T} + \mathbf{T}^T)}_{\text{symm}} + 0.5 \underbrace{(\mathbf{T} - \mathbf{T}^T)}_{\text{skew}} = \text{symm } \mathbf{T} + \text{skew } \mathbf{T} \quad (1.31)$$

- **迹**：张量的迹为标量（对角线元素的和）：

$$\text{tr } \mathbf{T} = T_{11} + T_{22} + T_{33} \quad (1.32)$$

- **对角**：由张量的对角线元素构成的矢量：

$$\text{diag } \mathbf{T} = (T_{11}, T_{22}, T_{33}) \quad (1.33)$$

- **dev、hyd**：每个二阶张量都可分解为 dev 部分（其迹为 0）和 hyd 部分（ $\mathbf{T} = s\mathbf{I}$ ， $s$  为标量）：

$$\mathbf{T} = \underbrace{\mathbf{T} - \frac{1}{3}(\text{tr } \mathbf{T})\mathbf{I}}_{\text{dev}} + \underbrace{\frac{1}{3}(\text{tr } \mathbf{T})\mathbf{I}}_{\text{hyd}} = \text{dev } \mathbf{T} + \text{hyd } \mathbf{T} \quad (1.34)$$

- **行列式**: 二阶张量的  $\det$  这样计算:

$$\det \mathbf{T} = \begin{vmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{vmatrix} = \frac{T_{11}(T_{22}T_{33} - T_{23}T_{32}) - T_{12}(T_{21}T_{33} - T_{23}T_{31}) + T_{13}(T_{21}T_{32} - T_{22}T_{31})}{1} = \frac{1}{6}e_{ijk}e_{pqr}T_{ip}T_{jq}T_{kr} \quad (1.35)$$

- **代数余子式**: 张量的余子式通常这样计算: 对于公式 1.35 中的  $3 \times 3$  行列式, 删除元素所在的行和列, 剩下元素取  $2 \times 2$  行列式, 例如  $T_{12}$  的余子式为:

$$\begin{vmatrix} T_{21} & T_{23} \\ T_{31} & T_{33} \end{vmatrix} = T_{21}T_{33} - T_{23}T_{31} \quad (1.36)$$

代数余子式是带符号的余子式, 符号这样计算:

$$\begin{cases} +, i+j \text{ 为偶数} \\ -, i+j \text{ 为奇数} \end{cases} \quad (1.37)$$

$\mathbf{T}$  的代数余子式即为:

$$\text{cof } \mathbf{T} = \frac{1}{2}e_{jkr}e_{ist}T_{sk}T_{tr} \quad (1.38)$$

- **逆**: 张量的逆这样计算:

$$\text{inv } \mathbf{T} = \frac{\text{cof } \mathbf{T}^T}{\det \mathbf{T}} \quad (1.39)$$

- **霍奇对偶**: 张量的霍奇对偶这样计算:

$$* \mathbf{T} = (T_{23}, -T_{13}, T_{12}) \quad (1.40)$$

### 1.3.10 标量算术

OpenFOAM 支持大部分的标量算术例如平方根、指数、对数、正弦余弦等, 它们在表 1.2 中列出。OpenFOAM 还提供了 3 个针对标量的附加函数即:

- **Sign**:

$$\text{sgn}(s) = \begin{cases} 1 & \text{如果 } s \geq 0 \\ -1 & \text{如果 } s < 0 \end{cases} \quad (1.41)$$

- **Positive**:

$$\text{pos}(s) = \begin{cases} 1 & \text{如果 } s \geq 0 \\ 0 & \text{如果 } s < 0 \end{cases} \quad (1.42)$$

- **Limit**:

$$\text{limit}(s, n) = \begin{cases} s & \text{如果 } s < n \\ 0 & \text{如果 } s \geq 0 \end{cases} \quad (1.43)$$

## 1.4 OpenFOAM 张量类

在 OpenFOAM 中，primitive 类包含了迄今为止所有的有关张量算术的计算。表 1.1 列举了 OpenFOAM 提供的基本的张量类型，表中也提供了访问张量元素的函数：

阶数	名字	基本类	访问函数
0	标量	scalar	
1	矢量	vector	x(), y(), z()
2	张量	tensor	xx(), xy(), xz()

表 1.1 OpenFOAM 张量基本类

在 OpenFOAM 中我们可以把张量：

$$T = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad (1.44)$$

这样定义：

```
tensor T(1, 2, 3, 4, 5, 6, 7, 8, 9);
```

我们可以利用 xz() 函数来访问  $T_{13}$  或者  $T_{xz}$ ，例如：

```
Info << "Txz = " << T.xz() << endl;
```

这会输出：

```
Txz = 3
```

### 1.4.1 OpenFOAM 张量操作

1.3 节描述的所有张量操作在 OpenFOAM 中都有特定的语法来实现，并且尽可能的和相对应的算符名称或图形相近。一些函数的名字直接使用操作符的名称，例如 symm()，其他的一些可以用操作符直接实现，例如 \*，所有的函数参见表 1.2：

操作	描述	数学形式	OpenFOAM 中的实现
加法		$\mathbf{a} + \mathbf{b}$	a + b
减法		$\mathbf{a} - \mathbf{b}$	a - b

乘以标量		$\mathbf{sa}$	$s * a$
除以标量		$\mathbf{a} / s$	$a / s$
外积	$\mathbf{a}$ 和 $\mathbf{b}$ 的阶数 $\geq 1$	$\mathbf{ab}$	$a * b$
内积	$\mathbf{a}$ 和 $\mathbf{b}$ 的阶数 $\geq 1$	$\mathbf{a} \cdot \mathbf{b}$	$a \& b$
双内积	$\mathbf{a}$ 和 $\mathbf{b}$ 的阶数 $\geq 2$	$\mathbf{a} : \mathbf{b}$	$a \&\& b$
叉乘	$\mathbf{a}$ 和 $\mathbf{b}$ 的阶数 $= 1$	$\mathbf{a} \times \mathbf{b}$	$a \wedge b$
平方		$\mathbf{a}^2$	$\text{sqr}(a)$
模的平方		$ \mathbf{a} ^2$	$\text{magSqr}(a)$
模		$ \mathbf{a} $	$\text{mag}(a)$
幂	$n = 0, 1, \dots, 4$	$\mathbf{a}^n$	$\text{pow}(a, n)$
元素均值	$i = 0, 1, \dots, N$	$\bar{a}_i$	$\text{cmptAv}(a)$
元素最大值	$i = 0, 1, \dots, N$	$\max(a_i)$	$\max(a)$
元素最小值	$i = 0, 1, \dots, N$	$\min(a_i)$	$\min(a)$
缩放		$\text{scale}(\mathbf{a}, \mathbf{b})$	$\text{scale}(a, b)$
<b>2 阶张量操作</b>			
转置		$\mathbf{T}^T$	$\text{T.T}()$
对角		$\text{diag}\mathbf{T}$	$\text{diag}(\text{T})$
迹		$\text{tr}\mathbf{T}$	$\text{tr}(\text{T})$
Dev		$\text{dev}\mathbf{T}$	$\text{dev}(\text{T})$
Sym		$\text{symm}\mathbf{T}$	$\text{symm}(\text{T})$
Skew-sym		$\text{skew}\mathbf{T}$	$\text{skew}(\text{T})$
行列式		$\det\mathbf{T}$	$\det(\text{T})$
代数余子式		$\text{cof}\mathbf{T}$	$\text{cof}(\text{T})$
逆		$\text{inv}\mathbf{T}$	$\text{inv}(\text{T})$
霍奇对偶		$*\mathbf{T}$	$*\text{T}$
<b>标量算术</b>			
取正负		$\text{sgn}(s)$	$\text{sign}(s)$
取正		$s \geq 0$	$\text{pos}(s)$
取负		$s < 0$	$\text{neg}(s)$
极限		$\lim(s, n)$	$\text{limit}(s, n)$
平方根		$\sqrt{s}$	$\text{sqr}(s)$
e		$\exp s$	$\exp(s)$
对数		$\ln s$	$\log(s)$
10 为基的对数		$\log_{10} s$	$\log_{10}(s)$
sin		$\text{asin } s$	$\sin(s)$
cos		$\cos s$	$\cos(s)$
tan		$\tan s$	$\tan(s)$
arcsin		$\text{asin } s$	$\text{asin}(s)$
arccos		$\text{acos } s$	$\text{acos}(s)$

arctan		atan $s$	atan( $s$ )
sinh		sinh $s$	sinh( $s$ )
cosh		cosh $s$	cosh( $s$ )
tanh		tanh $s$	tanh( $s$ )
arcsinh		asinh $s$	asinh( $s$ )
arccosh		acosh $s$	acosh( $s$ )
arctanh		atanh $s$	atanh( $s$ )
误差函数		erf $s$	erf( $s$ )
C 误差函数		erfc $s$	erfc( $s$ )
Gamma 对数函数		ln $\Gamma s$	lgamma( $s$ )
0 阶 1 型 Bessel 函数		$J_0 s$	j0( $s$ )
1 阶 1 型 Bessel 函数		$J_1 s$	j1( $s$ )
0 阶 2 型 Bessel 函数		$Y_0 s$	y0( $s$ )
1 阶 2 型 Bessel 函数		$Y_1 s$	y1( $s$ )
<b>a</b> 和 <b>b</b> 为任意阶数的张量（除非进行指定）， $s$ 为标量，N 为元素数			

表 1.2 OpenFOAM 中的张量操作

## 1.5 量纲单位

在连续介质里，物理量都具有量纲。例如质量：kg，体积：m<sup>3</sup>，压力：Pa。这些物理量的代数几何操作要求量纲保持一致。对于某些同样量纲的物理量，只有加、减、等于是具有物理意义的。为了防止对物理量进行无意义或者错误的运算，OpenFOAM 推荐用户将量纲依附于场数据以及某个物理量，这样在对其进行张量操作的时候，OpenFOAM 会执行量纲检查。

量纲我们用 dimensionSet 类来定义，例如：

```
dimensionSet pressureDims(1, -1, -2, 0, 0, 0, 0);
```

编号	名称	SI 单位	USCS 单位
1	质量	kg	lbm
2	米	m	ft
3	时间	s	s
4	开尔文	K	° R
5	摩尔质量	kgmol	lbmol
6	电流	A	A
7	光强	cd	cd

表 1.3 S.I. 量纲

表 1.3 列举了 OpenFOAM 植入的 S.I. 量纲制。我们上个范例中的 pressureDims 使用 dimensionSet 把单位设置为 kg m<sup>-1</sup> s<sup>-2</sup>，因为我们 pressureDims 序列中第一个值为 1，这对应 k<sup>1</sup>，第二

个值为-1，这对应  $m^{-1}$ ，以此类推。附带单位的张量使用 `dimensioned<Type>` 模板类来定义，其中 `type` 是 `scalar`, `vector`, `tensor` 等。`dimensioned<Type>` 将存储一个 `word` 类、一种 `<Type>` 类型、和 `dimensionSet`:

```
dimensionedTensor sigma
(
    "sigma",
    dimensionSet      (1, -1, -2, 0, 0, 0, 0),
    tensor            (1e6,0,0,0,1e6,0,0,0,1e6),
);
```

上述这段代码将会定义这样一个张量（应力单位）:

$$\sigma = \begin{pmatrix} 10^6 & 0 & 0 \\ 0 & 10^6 & 0 \\ 0 & 0 & 10^6 \end{pmatrix} \quad (1.45)$$

## 第二章

# 离散

目前我们处理的仅仅是张量算术。但是我们需要求解的是关于时间和空间的偏微分方程组。因此我们将拓展我们的张量为张量场，例如：一个关于时间和空间变化的张量。在这一章节我们首先对我们可能会遇到的偏微分算符进行描述。然后我们说明张量场在 OpenFOAM 中是如何构建的，以及如何将张量场离散成为方程组。

### 2.1 微分算符

在我们处理导数之前，我们首先通过索引符 $\partial_i$ 引入 nabla 算符 $\nabla$ ：

$$\nabla = \partial_i = \left( \frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3} \right) \quad (2.1)$$

nabla 算符在使用的时候遵循以下原则：

- 当 nabla 算符作用在乘积上的时候，其和乘积的导数相同。例如： $\partial_i ab = (\partial_i a)b + a(\partial_i b)$ ；
- 在其它情况下 nabla 算符的运算和一般的向量相同；

#### 2.1.1 梯度

如果定义一个标量场  $s$  并且  $s$  为连续可导的，我们有矢量场 $\nabla s$ ：

$$\nabla s = \partial_i s = \left( \frac{\partial s}{\partial x_1}, \frac{\partial s}{\partial x_2}, \frac{\partial s}{\partial x_3} \right) \quad (2.2)$$

梯度我们可以用在任何张量场上，得到的是更高一阶的张量场，例如，矢量场  $\mathbf{a}$  的梯度就是二阶张量场：

$$\nabla \mathbf{a} = \partial_i a_i = \begin{pmatrix} \partial a_1/\partial x_1 & \partial a_1/\partial x_2 & \partial a_1/\partial x_3 \\ \partial a_2/\partial x_1 & \partial a_2/\partial x_2 & \partial a_2/\partial x_3 \\ \partial a_3/\partial x_1 & \partial a_3/\partial x_2 & \partial a_3/\partial x_3 \end{pmatrix} \quad (2.3)$$

### 2.1.2 散度

如果矢量场  $\mathbf{a}$  是连续可导，那么  $\mathbf{a}$  的散度场为标量场：

$$\nabla \cdot \mathbf{a} = \partial_i a_i = \frac{\partial a_1}{\partial x_1} + \frac{\partial a_2}{\partial x_2} + \frac{\partial a_3}{\partial x_3} \quad (2.4)$$

散度可以在任何一阶以及以上阶张量场上操作，结果为低一阶的张量场。例如，二阶张量场  $\mathbf{T}$  的散度为矢量场：

$$\nabla \cdot \mathbf{T} = \partial_i T_{ij} = \begin{pmatrix} \partial T_{11}/\partial x_3 + \partial T_{12}/\partial x_3 + \partial T_{13}/\partial x_3 \\ \partial T_{21}/\partial x_3 + \partial T_{22}/\partial x_3 + \partial T_{23}/\partial x_3 \\ \partial T_{31}/\partial x_3 + \partial T_{32}/\partial x_3 + \partial T_{33}/\partial x_3 \end{pmatrix} \quad (2.5)$$

### 2.1.3 旋度

如果矢量场  $\mathbf{a}$  是连续可导，那么  $\mathbf{a}$  的旋度场为矢量场：

$$\nabla \times \mathbf{a} = e_{ijk} \partial_j a_k = \left( \frac{\partial a_3}{\partial x_2} - \frac{\partial a_2}{\partial x_3}, \frac{\partial a_1}{\partial x_3} - \frac{\partial a_3}{\partial x_1}, \frac{\partial a_2}{\partial x_1} - \frac{\partial a_1}{\partial x_2} \right) \quad (2.6)$$

旋度和散度的关系为：

$$\nabla \times \mathbf{a} = 2(* \text{skew} \nabla \mathbf{a}) \quad (2.7)$$

### 2.1.4 拉普拉斯

拉普拉斯操作在数学上可以看做是散度和梯度的组合，即： $\nabla^2 = \nabla \cdot \nabla$ 。然而，拉普拉斯应该看作为是单步操作：将张量场转化为同阶张量场的操作，而不是两步张量场操作的结合（先增加一阶然后再减去一阶）。

实际上，拉普拉斯最好定义为标量操作符的形式（正如我们定义  $\nabla$  操作符一样）：

$$\nabla^2 = \partial^2 = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} + \frac{\partial^2}{\partial x_3^2} \quad (2.8)$$

例如，对于一个标量场  $s$  的拉普拉斯场同样也为一个标量场：

$$\nabla^2 s = \partial^2 s = \frac{\partial^2 s}{\partial x_1^2} + \frac{\partial^2 s}{\partial x_2^2} + \frac{\partial^2 s}{\partial x_3^2} \quad (2.9)$$

### 2.1.5 时间导数

存在不止一种有关张量场的时间导数定义。为了准确的描述时间导数我们必须首先知道张量是和一个移动的某个体积下的物理量有关的。如果我们追踪这个无限小的微元（或者微粒），它是即时移动的并且物理张量  $\phi$  会跟随时间而发生变化，我们定义物质导数为<sup>2</sup>：

<sup>2</sup> 拉格朗日方法



$$\frac{D\phi}{Dt} = \lim_{\Delta t \rightarrow 0} \frac{\Delta\phi}{\Delta t} \quad (2.10)$$

在连续介质问题中，特别是流体力学中，我们通常在某个固定点来观测某个物理张量跟随时间的变化形式<sup>3</sup>。我们用  $\partial/\partial t$  来表示在空间的某个点下的时间导数项，我们用它可以表示物质导数：

$$\frac{D\phi}{Dt} = \frac{\partial\phi}{\partial t} + \mathbf{U} \cdot \nabla\phi \quad (2.11)$$

其中  $\mathbf{U}$  是某个物理量  $\phi$  的速度，2.11 公式右边的第二项我们称为  $\phi$  的对流率。

## 2.2 离散简述

离散的意思就是用一系列离散的量来近似代表一系列的连续值。有限体积法以及其他的例如有限元、有限差分在离散的时候都遵循下面的原则：

- **空间离散：**通过一系列的点来定义求解域，它填充整个空间并且是有界的；
- **时间离散：**对于瞬态问题，将计算时间分为有限的小的时间间隔；
- **方程离散：**针对某个物理问题特定的偏微分方程，生成一系列的代数方程来代表在空间离散后的物理量；

### 2.2.1 OpenFOAM 链表和场

OpenFOAM 需要存储大量的数据使用函数对数据进行大量的操作。因此 OpenFOAM 提供模板类 `List<Type>`，其可以通过类模板创建一系列的 `Type` 对象来继承其特性。例如，`vector` 的 `List` 即为 `List<vector>`。

张量链表在 OpenFOAM 中通过 `Field<Type>` 来指定。为了使代码更加的清晰易懂，所有的 `Field<Type>` 类型，例如 `Field<vector>`，都使用 `typedef` 来重新定义为 `scalarField`, `vectorField`, `tensorField`, `symmTensorField`, `tensorThirdField` 以及 `symmTensorThirdField`。张量场上的代数算术有一些显而易见的数学规则，例如张量场的元素数量必须一致等。OpenFOAM 也支持单一场和一个数值的操作。例如，所有的速度场都乘以标量 2，即： $\mathbf{U} = 2.0 \times \mathbf{U}$ 。

## 2.3 计算域离散

计算域的离散参考图 2.1 所示。首先我们将整个计算空间离散为网格，在这个计算网格基础上进行偏微分方程的离散。如果时间也需要进行离散的话是非常简单的。将时间划分为一系列的时间步  $\Delta t$  (可以随着计算而改变大小) 即可。时间步长一般和计算中的某些条件相对应。

更详细来讲，空间离散需要将整个计算域划分为非常多小的网格（有限控制体）。这些

<sup>3</sup> 欧拉方法

网格互相是连续相连的，例如他们不能互相重合，并且他们需要填充整个计算域。典型的空  
间离散后的网格如图 2.1 所示：

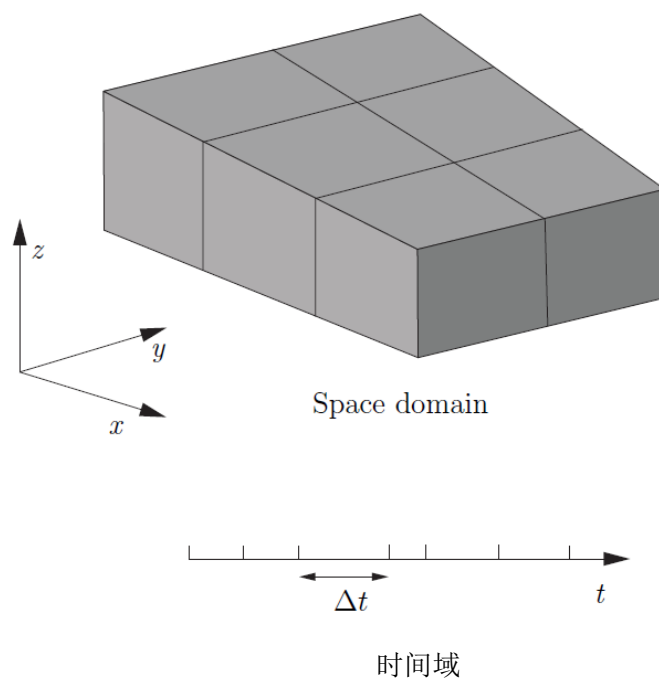


表 2.1 计算域离散

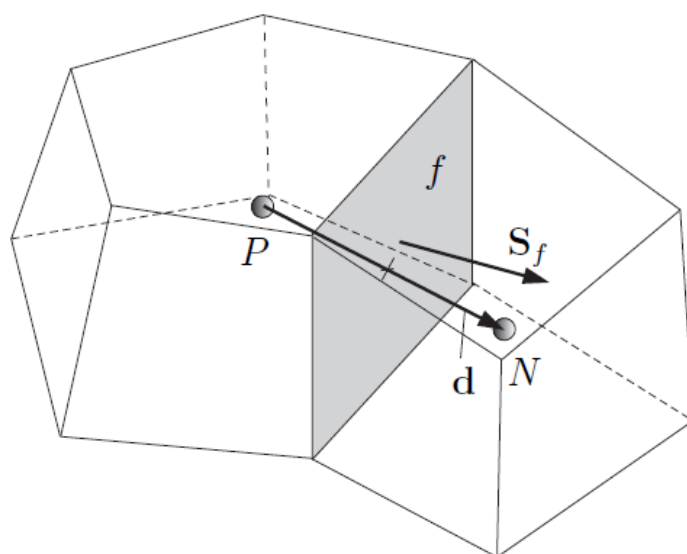


表 2.2 有限体积离散的相关参数

即使某些物理量以及其他的相关参数需要储存在面或者顶点上，他们大部分都储存在网格中心  $P$ 。网格单元通过一系列的平面来界定，我们用  $f$  来表示面。在 OpenFOAM 中，网格单元面的数量是没有限制的，这种网格我们通常称之为任意非结构网格。这种叫法可以和某些网格（这些网格的网格单元面和坐标轴相依赖）来进行区分。使用这种网格来进行计算的计算流体力学代码允许非常灵活的网格生成和操作。这在几何体非常复杂以及需要处理动网格的时候优势明显。

虽然大部分物理量存储在网格中心，但是一些物理量依然存储在网格单元面，在 OpenFOAM 中具有两种网格单元面：

- **内部面：**连接两个网格单元的面（且仅仅连接两个网格单元）。对于每个内部面，OpenFOAM 将其连接的两个网格单元一个指定为 *owner*，一个指定为 *neighbor*；
- **边界面：**从属一个网格单元的面，且和计算域边界重合。这些面只有一个 *owner* 网格单元；

### 2.3.1 定义 OpenFOAM 网格

OpenFOAM 中使用不同等级来进行网格描述，最基本的网格类为 `polyMesh`，之所以叫这个名字是因为它建立在多面体基础之上。描述 `polyMesh` 需要使用下面的相关信息来定义几何和网格，详见图 2.3：

- **Points：**网格单元点位置矢量链表（`vectorField`），使用 `typedef` 定义后的名称为 `pointField`；
- **Faces：**一系列网格单元面列表 `List<face>`，也即 `faceList`，其中 `face` 类通过 `pointField` 的顶点标号相关信息来定义；
- **Cells：**一系列网格单元列表 `List<cell>`，也即 `cellList`，其中 `cell` 类通过 `faceList` 的面标号相关信息来定义；
- **Boundary：**`polyBoundaryMesh` 分解为一系列的 `patch`，`polyPatchList` 代表不同的边界区域。边界依靠这种方式来进行区分这样可以使用不同的边界条件。所有的 `polyPatch` 面都作为一个单一 `block` 存储在 `faceList` 中，这样他的面可以使用 `slice` 类（存贮 `block` 的第一个面和最后一个面的引用）来访问。每个 `polyPatch` 需要从下面两个信息来组建：
  - ✓ `slice`：见上文；
  - ✓ `word`：名字；

有限体积离散需要使用存储在 `polyMesh` 上的几何信息。因此 OpenFOAM 把 `polyMesh` 拓展为 `fvMesh` 类，其存储用于有限体积离散的相关信息。参见图 2.1，`fvMesh` 从 `polyMesh` 来组建，`fvMesh` 可以在运行的时候进行更新（例如进行网格细化、动网格的算例）。

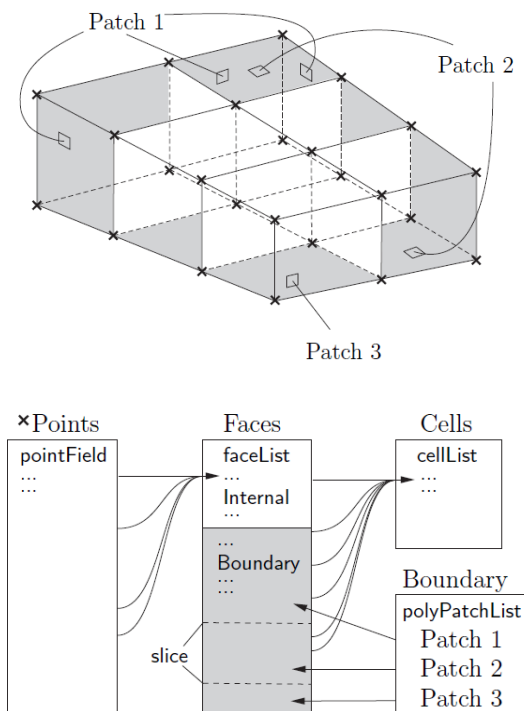


图 2.3 OpenFOAM 的网格描述纲要

### 2.3.2 定义 geometricField

目前为止我们定义了场（张量链表）和网格。他们可以通过在我们的计算域定义张量场来结合起来，这通过 OpenFOAM 中的模板类 `geometricField<Type>` 来实现。Field 的值区分为计算域的内部值（计算域内部网格单元中心的值）和边界值（计算域的边界面的值）。`geometricField<Type>` 存储下列信息：

- **Internal field:** 简单的 `Field<Type>` 类型，详见 2.2.1 节；
- **Boundary field:** 一种 `GeometricBoundaryField`，需要为每个 patch 的每个网格单元面定义 Field。于是这就是场的场，存储在 `FieldField<Type>` 里，`fvBoundaryMesh` 引用也存储在此处；
- **Mesh:** `fvMesh` 的引用，其会添加一下附加信息，例如场信息存储在网格单元中心还是面上；
- **Dimensions:** 一个 `dimensionSet` 类型，详见《OpenFOAM 用户指南》4.2.6 节；
- **Old values:** 在时间离散中上一个时间步的场信息。`geometricField<Type>` 会存储上一个时间步（过去的时间步，甚至上上个时间步）的引用信息；
- **Previous iteration values:** 在使用低松弛求解的时候需要存储之前迭代的信息。因此，如果需要的话，`geometricField<Type>` 会存储之前迭代的所需信息；

类	描述	符号	成员函数
volScalarField	网格体积 <sup>4</sup>	$V$	V()
surfaceVectorField	面矢量	$\mathbf{S}_f$	Sf()
surfaceScalarField	面大小	$ \mathbf{S}_f $	magSf()
volVectorField	网格中心	$\mathbf{C}$	C()
surfaceVectorField	面心	$\mathbf{C}_f$	Cf()
surfaceScalarField	面通量	$\phi_g$	phi()

表 2.1 fvMesh 存储的信息

正如 2.3 节我们讨论的，我们主要将物理量定义在网格单元中心，一部分定义在网格面，很少的一部分我们定义在网格单元顶点上。geometricField<Type>也使用 typedef 进行了重新命名来表示定义场的位置信息：

- volField<Type>：网格单元中心场；
- surfaceField<Type>：网格单元面场；
- pointField<Type>：网格单元顶点场；

这些通过 typedef 重新命名的 geometricField<Type>请查阅图 2.4，geometricField<Type>继承了所有的 Field<Type>数学操作，并且可以使用 dimensionSet 进行量纲检查。它也可以使用下述章节的有限体积离散的相关函数。geometricField<Type>的类结构请参考图 2.5<sup>5</sup>。

## 2.4 方程离散

偏微分方程组可以转化为一系列的代数方程组，如下所示：

$$[A][x] = [b] \quad (2.12)$$

其中[A]是方阵，[x]是相关物理量的未知矢量，[b]是源项矢量。我们把[x]和[b]称之为矢量不是因为他们是具有三个元素的矢量，而是他们是一个和矩阵相关的专业词汇。他们是定义在计算域的一系列的值，例如一个 geometricField<Type>类型，在使用有限体积离散的时候一般为 volField<Type>类型。

[A]是代数方程组的系数矩阵，它不能用 geometricField<Type>来描述。因此它们使用他们自己的类进行描述：fvMatrix。fvMatrix<Type>通过对 geometricField<Type>离散进行创建因此会继承<Type>的特性。他们可以使用常规的代数方程组操作例如加法，减法，乘法之类。

在 OpenFOAM 中，偏微分方程中的每个项都需要使用静态函数类 finiteVolumeMethod 以及 finiteVolumeCalculus 进行表示，简化的写法即为：fvm，fvc。

<sup>4</sup> 此列并不是第一列的翻译

<sup>5</sup> 这个图表并不是 geometricField<Type>的完整描述，在这里只是简单的介绍 geometricField<Type>的主要衍生类

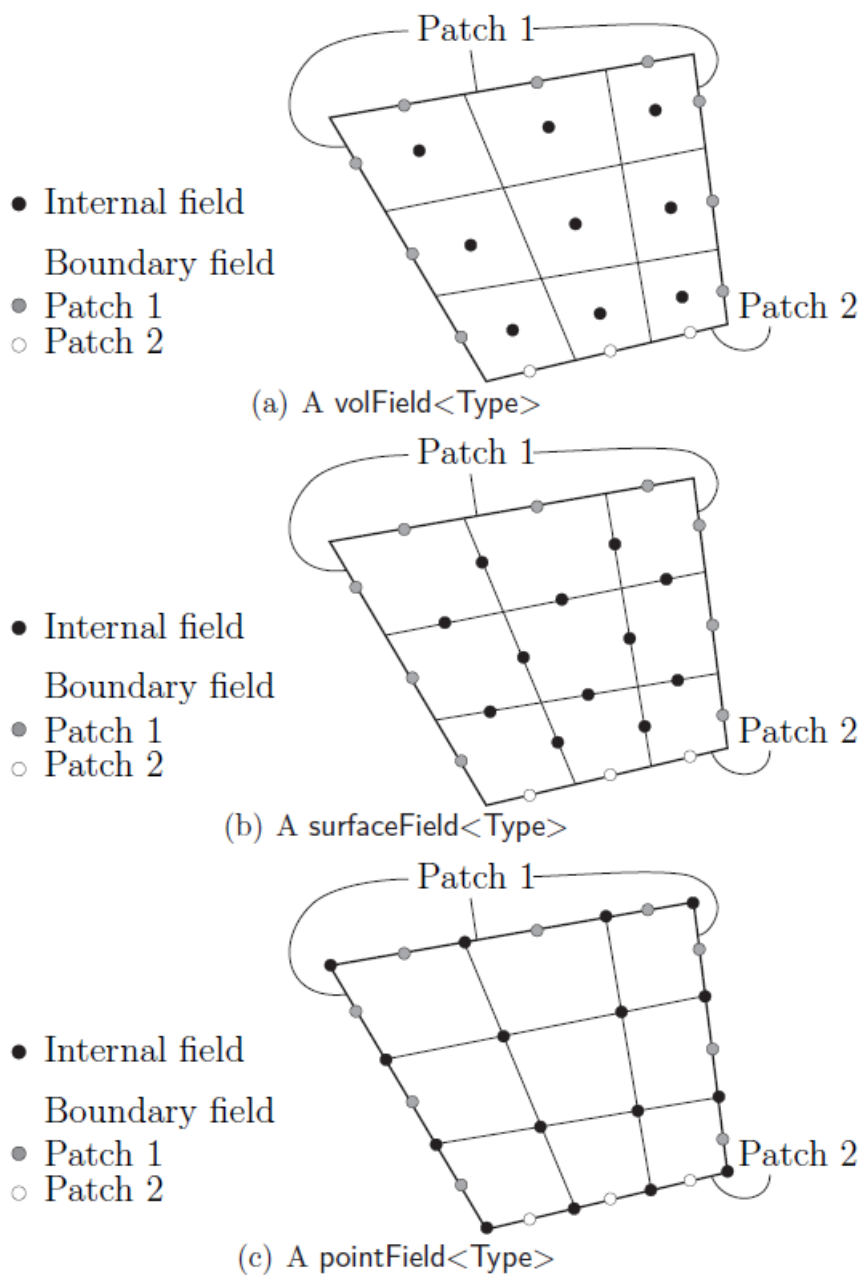
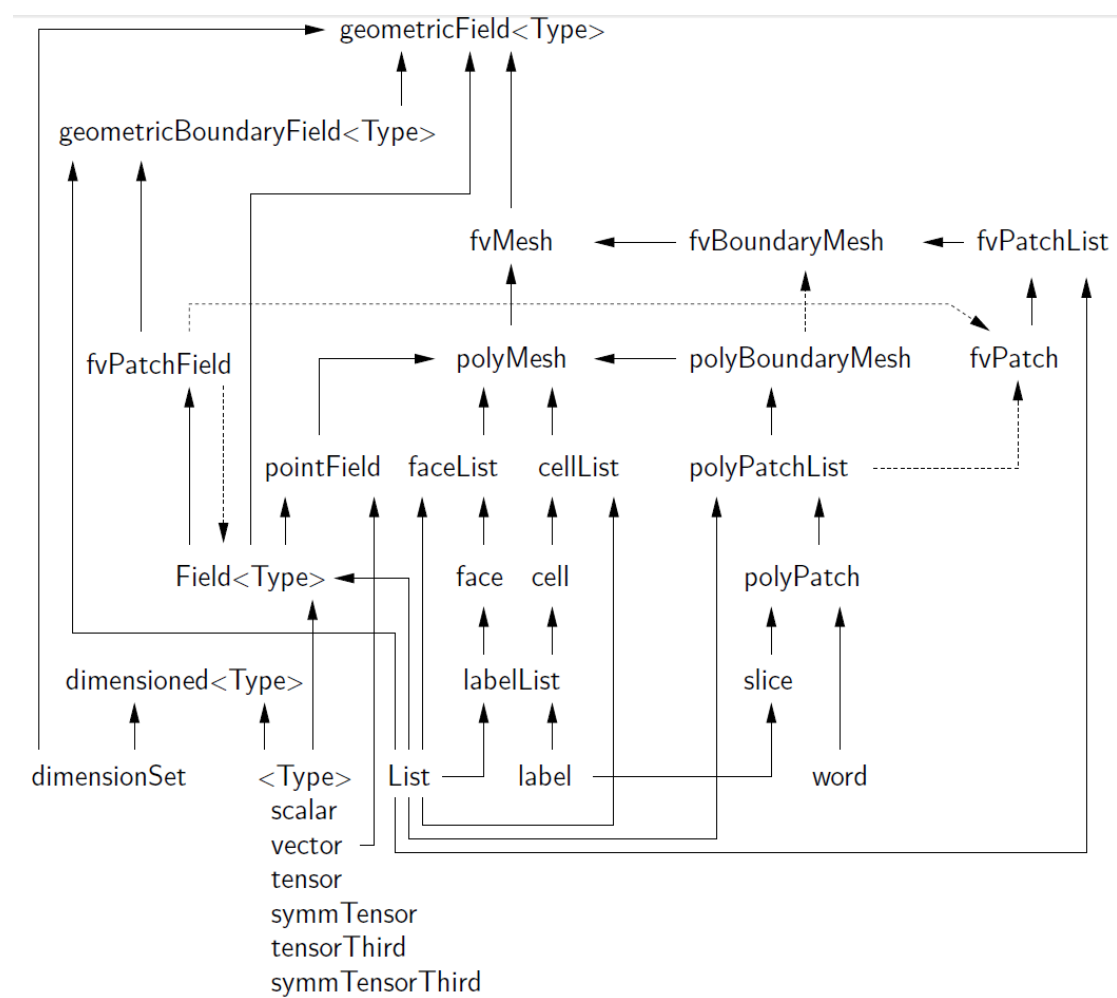


图 2.4 定义在 2 个边界 patch 上的 geometricField<Type>类型  
 (使用 2 维几何进行简要描述)

图 2.5 `geometricField<Type>`的基本类结构

fvm 和 fvc 都相应的包含各自的函数以代表相对应的微分算符，例如  $\nabla^2$ ， $\nabla \cdot$ ， $\partial/\partial t$ ，我们区分 fvm 和 fvc 的目的在于：

- fvm 对偏微分方程的项进行隐式离散，返回一个 fvMatrix<Type>；
- fvc 的某些函数计算显性导数以及一些其他的显性操作，返回一个 geometricField - <Type>；

图 2.6 即为定义在网格上的 geometricField<Type>，其有两个边界 patch，为了简化说明如何将一个类型转化为另一个类型我们使用 2 维图形进行表示：

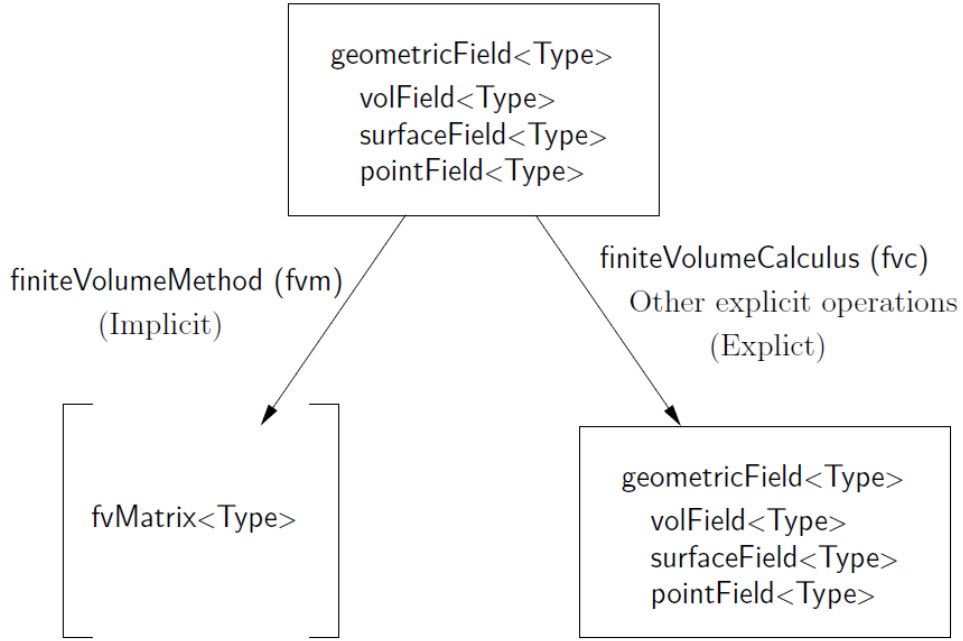


图 2.6 geometricField<Type>及其数学操作

表 2.2 列举了用于进行偏微分方程离散的 fvm 和 fvc 可用的大部分函数。在进行有限体积离散的时候，每个偏微分方程的项首先需要在有限控制体  $V$  内进行积分。大部分空间导数项都使用高斯定律来将体积分转化为面积分：

$$\int_V \nabla \otimes \phi \, dV = \int_S d\mathbf{S} \otimes \phi \quad (2.13)$$

其中  $\mathbf{S}$  是面矢量， $\phi$  表示任意的张量场， $\otimes$  表示任意的张量积，例如内积、外积、叉乘。相对应的就是散度： $\nabla \cdot \phi$ ，梯度  $\nabla \phi$ ，以及  $\nabla \times \phi$ 。在积分之后，我们需要使用合适的格式来进行线性化，这将在后续章节有所提及。OpenFOAM 提供了很多种格式进行离散，但某些项经常使用一种格式来离散。离散格式的选择可以在代码中直接指定，或者可以从输入文件中指定，他们存储在 fvSchemes 类中。



离散项	显性/隐性	数学形式	代码实现
拉普拉斯项	Imp/Exp	$\nabla^2 \phi$ $\nabla \cdot \Gamma \nabla \phi$	laplacian(phi) laplacian(Gamma, phi)
时间导数	Imp/Exp	$\frac{\partial \phi}{\partial t}$ $\frac{\partial \rho \phi}{\partial t}$	ddt(phi) ddt(rho, phi)
二阶时间导数	Imp/Exp	$\frac{\partial}{\partial t} \left( \rho \frac{\partial \phi}{\partial t} \right)$	d2dt2(rho, phi)
对流项	Imp/Exp	$\nabla \cdot (\psi)$ $\nabla \cdot (\psi \psi)$	div(psi, scheme)* div(psi, phi, word)* div(psi, phi)
散度项	Exp	$\nabla \cdot \chi$	div(chi)
梯度项	Exp	$\nabla \chi$ $\nabla \phi$	grad(chi) gGrad(phi) lsGrad(phi) snGrad(phi) snGradCorrection(phi)
梯度梯度项	Exp	$ \nabla \nabla \phi ^2$	sqrGradGrad(phi)
旋度项	Exp	$\nabla \times \phi$	curl(phi)
源项	Imp Imp/Exp	$\rho \phi$	Sp(rho, phi) SuSp(rho, phi)
<p>fvm::SuSp 通过判断 rho 的符号进行显性或者隐性离散；  显性项离散可以通过 vol&lt;Type&gt;Field 引入，例如 rho*phi；  函数参数可以为以下类型：  phi: vol&lt;Type&gt;Field;  Gamma: scalar volScalarField, surfaceScalarField, volTensorField, surfaceTensorField;  rho: scalar, volScalarField;  psi: surfaceScalarField;  chi: surface&lt;Type&gt;Field, vol&lt;Type&gt;Field;</p>			

表 2.2 OpenFOAM 偏微分方程离散

### 2.4.1 拉普拉斯项

在有限控制体内，拉普拉斯项这样积分以及线性化：

$$\int_V \nabla \cdot (\Gamma \nabla \phi) dV = \int_S d\mathbf{S} \cdot (\Gamma \nabla \phi) = \sum_f \Gamma_f \mathbf{S}_f \cdot (\nabla \phi)_f \quad (2.14)$$

当网格单元中心  $\mathbf{P}$  和相邻网格单元中心  $\mathbf{C}$  的长度矢量  $\mathbf{d}$  和这两个网格单元的内部面正交的时候，即  $\mathbf{d}$  平行于  $\mathbf{S}_f$  的时候，面梯度这样计算<sup>6</sup>：

$$\mathbf{S}_f \cdot (\nabla \phi)_f = |S_f| \frac{\phi_N - \phi_P}{|\mathbf{d}|} \quad (2.15)$$

在网格非正交的情况下，我们需要引入一个显性项<sup>7</sup>。

### 2.4.2 对流项

在有限控制体内，对流项这样积分以及线性化：

$$\int_V \nabla \cdot (\rho \mathbf{U} \phi) dV = \int_S d\mathbf{S} \cdot (\rho \mathbf{U} \phi) = \sum_f \mathbf{S}_f \cdot (\rho \mathbf{U})_f \phi_f = \sum_f F \phi_f \quad (2.16)$$

面值  $\phi_f$  可以使用一系列的格式进行计算：

- **中心差分 (CD)**：无界二阶精度：

$$\phi_f = f_x \phi_P + (1 - f_x) \phi_N \quad (2.17)$$

其中  $f_x = \overline{fN}/\overline{PN}$ ， $\overline{fN}$  是面  $f$  和网格单元中心  $N$  的距离， $\overline{PN}$  是网格单元中心  $P$  和  $N$  的距离；

- **迎风格式 (UD)**：牺牲精度来确保有界，依据流向决定  $\phi_f$  的值：

$$\phi_f = \begin{cases} \phi_P & \text{for } F \geq 0 \\ \phi_N & \text{for } F < 0 \end{cases} \quad (2.18)$$

- **混合格式 (BD)**：即中心差分和迎风格式的混合，力图在合理的精度下保证有界：

$$\phi_f = (1 - \gamma)(\phi_f)_{UD} + \gamma(\phi_f)_{CD} \quad (2.19)$$

OpenFOAM 植入了 Gamma 差分格式，他们可以选择混合系数  $\gamma$ ，同样，Open-FOAM 也提供了一些其他的著名格式例如 vanLeer，SUPERBEE，MINMOD 等；

<sup>6</sup> 此公式类似一种格式

<sup>7</sup> 请参考 Hrv 的博士论文

### 2.4.3 一阶时间导数

在有限控制体内，一阶时间导数项这样积分：

$$\frac{\partial}{\partial t} \int_V \rho \phi dV \quad (2.20)$$

这一项需要将时间进行简单的离散，在这里我们引进几个项：

- **新值**： $\phi^n = \phi(t + \Delta t)$ ，即我们在当前时间步需要要求的物理量；
- **旧值**： $\phi^0 = \phi(t)$ ，我们在上一个时间步存储的物理量；
- **旧-旧值**： $\phi^{00} = \phi(t - \Delta t)$ ，我们在上上一个时间步存储的物理量；

一阶或二阶时间导数项的离散可以在相关的字典文件中通过 `timeScheme` 关键词来进行定义，详情请参考《OpenFOAM 用户指南》4.4 节；

- **欧拉显性**：这样指定：`timeScheme EulerImplicit`，其为一阶精度：

$$\frac{\partial}{\partial t} \int_V \rho \phi dV = \frac{(\rho_P \phi_P V)^n - (\rho_P \phi_P V)^0}{\Delta_t} \quad (2.21)$$

- **向后差分**：这样指定：`timeScheme BackwardDifferencing`，其为需要存储旧-旧值的二阶精度格式，因此相对于欧拉显性需要更大的内存空间：

$$\frac{\partial}{\partial t} \int_V \rho \phi dV = \frac{3(\rho_P \phi_P V)^n - 4(\rho_P \phi_P V)^0 + (\rho_P \phi_P V)^{00}}{2\Delta_t} \quad (2.22)$$

### 2.4.4 二阶时间导数项

在有限控制体内，二阶时间导数项这样积分（其为一阶精度）：

$$\frac{\partial}{\partial t} \int_V \rho \frac{\partial \phi}{\partial t} dV = \frac{(\rho_P \phi_P V)^n - 2(\rho_P \phi_P V)^0 + (\rho_P \phi_P V)^{00}}{\Delta_t^2} \quad (2.23)$$

### 2.4.5 散度项

这一节我们所说的散度为显性项（以区别 2.4.2 所提及的对流项），这一节的散度项并不是速度项和某个物理量的乘积。在有限控制体内，散度项这样积分：

$$\int_V \nabla \cdot \phi dV = \int_S d\mathbf{S} \cdot \phi = \sum_f \mathbf{S}_f \cdot \phi_f \quad (2.24)$$

其中 `fvc::div` 的参数可以为 `surface<Type>Field` (需要指定  $\phi_f$ )，或者 `vol<Type>Field`，其需要使用 2.4.10 节提及的中心差分格式来进行面插值。

## 2.4.6 梯度项

梯度项是可以通过很多方式进行计算的显性项。格式可以这样指定：要么通过选择某个格式的梯度函数来计算，例如 `fvc::gGrad`，`fvc::lsGrad`，要么通过 `fvc::grad` 以及输入文件的 `grad - Scheme` 来计算。

- **高斯积分：**通过 `fvc::grad` 函数（并在 `gradScheme` 里面指定 `Gauss`）指定，或者使用 `fvc::gGrad` 函数指定。离散的时候需要在有限控制体内使用高斯定律进行积分：

$$\int_V \nabla \phi \, dV = \int_S d\mathbf{S} \phi = \sum_f \mathbf{S}_f \phi_f \quad (2.25)$$

和 `fvc::div` 函数一样，高斯积分的 `fvc::grad` 函数的参数可以为 `surface<Type>Field` 或者 `vol<Type>Field`。

- **最小二乘法：**建立在下述规则基础之上：
  1. 从点 P 的值，使用 P 点的梯度可以推算 N 点的值；
  2. N 点的推断值可以和 N 点的实际值进行比较，其值的不同即为误差；
  3. 如果我们使用某种梯度使得 P 节点和临点 N 的梯度的平方和最小，我们认为这个梯度假定是精准的；

最小二乘法可以通过 `fvc::grad` 函数（并在 `gradScheme` 里面指定 `leastSquares`）指定，或者使用 `fvc::lsGrad` 函数指定。离散首先计算每个 P 点的张量 **G**

$$\mathbf{G} = \sum_N \omega_N^2 \mathbf{d} \mathbf{d} \quad (2.26)$$

**d** 是 P 点到 N 点的矢量，且权函数  $\omega_N = 1/|\mathbf{d}|$ 。梯度可以这样计算：

$$(\nabla \phi)_P = \sum_N \omega_N^2 \mathbf{G}^{-1} \cdot \mathbf{d} (\phi_N - \phi_P) \quad (2.27)$$

- **面法相梯度：**面法向梯度  $\mathbf{n}_f \cdot (\nabla \phi)_f$  可以使用下面的公式计算：

$$(\nabla \phi)_f = \frac{\phi_N - \phi_P}{|\mathbf{d}|} \quad (2.28)$$

梯度项通过 `fvc::snGrad` 调用并返回一个 `surfaceField<Type>`，和 2.4.1 节的拉普拉斯离散一样， $\nabla \phi_f$  使用上述公式进行计算。同样的，在非正交网格中，面法相梯度在计算的时候我们引入一个线性的修正项。在代码中我们可以调用 `fvc::snGrad - Correction`。

### 2.4.7 梯度-梯度的平方

梯度-梯度的平方我们这样计算：首先对场取散度；再对这个取散度后的场继续取散度；然后计算结果的模的平方。在数学上，梯度梯度的平方这样表示： $|\nabla(\nabla\phi)^2|$ 。

### 2.4.8 旋度

旋度可以从梯度计算得来，参见 2.4.6 节。首先我们计算梯度，然后通过下面这个公式（公式 2.7）来计算旋度：

$$\nabla \times \phi = 2 * (\text{skew} \nabla \phi)$$

### 2.4.9 源项

源项的指定有三种方法：

- **显性：**每个显性的源项为一个 `volField<Type>`。因此，显性的源项在偏微分方程中可以简单的看做一个值。例如，如果我们打算求解泊松方程  $\nabla^2 \phi = f$ ，我们可以定义 `phi` 和 `f` 为一个 `volScalarField`，然后使用下面的代码：

```
solve(fvm::laplacian(phi) == f)
```

- **隐性：**隐性源项通过下面的公式在有限控制体内进行积分并线性化：

$$\int_V \rho \phi dV = \rho_P V_P \phi_P \quad (2.29)$$

- **显性/隐性：**隐性源项会改变矩阵的系数。根据源项的正负以及矩阵系数，隐形源项会使得矩阵的对角占优有所改变。如果降低矩阵的对角占优会在矩阵迭代求解的时候引入不稳定性。因此，OpenFOAM 提供了混合的源项离散形式：当源项小于 0 的时候进行隐性离散，源项大于 0 的时候进行显性离散<sup>8</sup>。从数学上来讲，节点 P 的矩阵系数是  $V_P \max(\rho_P, 0)$ ，源项为： $V_P \min(\rho_P, 0)$ ；

### 2.4.10 其他显性离散

我们还可以使用其他的离散方法在 `volField<Type>` 和 `surface<Type>Fields` 之间互相转换：

- **面积分：**`fvc::surfaceIntegrate` 其将面上的 `surface<Type>Field` 进行加和，然后除以网格单元体积： $(\sum_f \phi_f) V_p$ ，其返回一个 `volField<Type>`；
- **面加和：**`fvc::surfaceSum` 其将面上的 `surface<Type>Field` 进行加和： $(\sum_f \phi_f)$ ，其返回一个 `volField<Type>`；

<sup>8</sup> 原文有误

- **平均:** `fvc::average` 计算 `surface<Type>Field` 的面平均值:  $(\sum_f S_f \phi_f) / \sum_f S_f$ , 其返回一个 `volField<Type>`;
- **面插值:** `faceInterpolate()` 将 `volField<Type>` 从网格单元中心的值插值到面上并返回一个 `surface<Type>Field` (使用中心差分);

## 2.5 时间离散

虽然我们在 2.4.3 节和 2.4.4 节讨论了时间项的离散, 我们仍然需要讨论一下瞬态问题下的时间项处理问题。首先我们使用  $\mathcal{A}$  表示任意的张量算符 (例如拉普拉斯), 我们假定有下面的积分形式的偏微分方程:

$$\int_t^{t+\Delta t} \left[ \frac{\partial}{\partial t} \int_V \rho \phi dV + \int_V \mathcal{A} \phi dV \right] dt = 0 \quad (2.30)$$

参考公式 2.21 使用欧拉隐形, 第一项可以表示为:

$$\int_t^{t+\Delta t} \left[ \frac{\partial}{\partial t} \int_V \rho \phi dV \right] dt = \int_t^{t+\Delta t} \frac{(\rho_P \phi_P V)^n - (\rho_P \phi_P V)^o}{\Delta t} dt = \frac{(\rho_P \phi_P V)^n - (\rho_P \phi_P V)^o}{\Delta t} \Delta t \quad (2.31)$$

第二项可以表示为:

$$\int_t^{t+\Delta t} \left[ \int_V \mathcal{A} \phi dV \right] dt = \int_t^{t+\Delta t} \mathcal{A}^* \phi dt \quad (2.32)$$

其中  $\mathcal{A}^*$  表示  $\mathcal{A}$  的离散形式, 因此, 时间项的积分可以采用以下三种方式:

- **欧拉隐性:** 采用隐形式离散空间项, 因此在计算中取新值  $\phi^n$ :

$$\int_t^{t+\Delta t} \mathcal{A}^* \phi dt = \mathcal{A}^* \phi^n \Delta t \quad (2.33)$$

其为关于时间的一阶精度, 因此为有界且无条件稳定的;

- **显性:** 采用显形式离散空间项, 因此在计算中我们去旧值  $\phi^o$ :

$$\int_t^{t+\Delta t} \mathcal{A}^* \phi dt = \mathcal{A}^* \phi^o \Delta t \quad (2.34)$$

其为关于时间的一阶精度, 如果库郎数大于 1, 其会不稳定。库郎数这样定义:

$$C_o = \frac{\mathbf{U}_f \cdot \mathbf{d}}{|\mathbf{d}|^2 \Delta t} \quad (2.35)$$

其中  $\mathbf{U}_f$  为特征速度, 例如流体速度等;

- **C-N 格式**：使用显性隐性的混合来离散空间项，因此需要对新值和旧值取平均：

$$\int_t^{t+\Delta t} \mathcal{A}^* \phi dt = \mathcal{A}^* \left( \frac{\phi^n + \phi^o}{2} \right) \Delta t \quad (2.36)$$

其为关于时间的二阶精度，无条件稳定但是不一定有界；

### 2.5.1 OpenFOAM 中时间离散的代码

目前，对时间离散的处理要根据空间导数的处理方法（显性隐性）来选定。例如，我们打算求解下面这个瞬态扩散方程：

$$\frac{\partial \phi}{\partial t} = \kappa \nabla^2 \phi \quad (2.37)$$

欧拉隐性代码如下：

```
solve(fvm::ddt(phi) == kappa*fvm::laplacian(phi))
```

其中我们使用 `fvm` 类将拉普拉斯项隐性离散，如果显性离散的话即为下述：

```
solve(fvm::ddt(phi) == kappa*fvc::laplacian(phi))
```

其中我们使用 `fvc` 类将拉普拉斯项显性离散，C-N 格式的离散可以这样植入：

```
solve
(
    fvm::ddt(phi)
    ==
    kappa*0.5*(fvm::laplacian(phi) + fvc::laplacian(phi))
)
```

## 2.6 边界条件

边界条件对于我们需要计算的问题是非常重要的。因此我们在所有的边界面指定边界条件。边界条件分为两种类型：

- **Dirichlet**：描述边界上的变量值，因此我们称为固定值边界条件；
- **Neumann**：描述变量在边界上的法向梯度值，因此我们称之为固定梯度边界条件；

当我们在进行离散的时候，有时会包含这一项： $\Sigma$ ，因此我们需要考虑如果某个网格单元面为边界面的时候会发生什么样的情况：

- **Fixed value:** 我们在边界处指定固定值:

- ✓ 使用边界值 $\phi_b$ 代替 $\phi_f$ , 可参见方程 2.16;
- ✓ 如果需要计算面梯度 $(\nabla\phi)_f$ , 例如拉普拉斯项, 我们可以这样用公式表示:

$$\mathbf{S}_f \cdot (\nabla\phi)_f = |\mathbf{S}_f| \frac{\phi_b - \phi_p}{|\mathbf{d}|} \quad (2.38)$$

- **Fixed gradient:** 固定梯度边界条件需要计算梯度和边界面法向的内积:

$$g_b = \left( \frac{\mathbf{s}}{|\mathbf{s}|} \cdot \nabla\phi \right)_f \quad (2.39)$$

- ✓ 当需要边界的 $\phi_f$ 值的时候, 我们需要从网格单元中心值进行插值:

$$\phi_f = \phi_p + \mathbf{d} \cdot (\nabla\phi)_f = \phi_p + |\mathbf{d}|g_b \quad (2.40)$$

- ✓ 当需要面法向梯度的时候,  $\phi_b$ 可以直接进行替换:

$$\mathbf{S}_f \cdot (\nabla\phi)_f = |\mathbf{S}_f|g_b \quad (2.41)$$

### 2.6.1 物理边界条件

工程师们通常通过实际的物理情况来指定边界条件。相比于前述的各种公式, 真实的边界条件非常具有描述性。在不可压缩流中我们有下面几个具有代表性的物理边界条件:

- **进口:** 出口的速度场需要指定, 压力应该指定法向梯度为零;
- **出口 t:** 出口的压力场需要指定, 速度应该指定法向梯度为零;
- **无滑移壁面:** 壁面处的速度为 0, 因此需要指定速度固定值边界条件。压力指定为法向梯度为零 (因为壁面出的通量为 0);

有时, 计算域以及边界条件会呈对称分布, 我们只需要计算其中的一半区域就可以了。其他区域我们考虑为对称面。其中对称的交界面我们必须指定为:

- **对称平面:** 对称平面边界条件指定平面的法向梯度为零;



## 第三章

# 算例演示

在这一章中，我们讨论几个 OpenFOAM 所提供的算例。OpenFOAM 对每一个求解器都提供了对应的算例，有些算例在《OpenFOAM 用户指南》中第二章里进行了介绍。OpenFOAM 提供这些算例的目的在于帮助用户了解 OpenFOAM 求解器和其中的一些工具的特点，比如前处理和后处理工具以及数值格式与数值算法。这些算例的运行结果也证明了各个求解器的正确性。

每个算例都从如下几个方面来描述所要求解的问题：几何模型、初始条件、边界条件、所解控制方程、求解过程中用到的模型和物理参数。这些例子中所选用的计算域有可能只是原几何模型的一部分（比如当我们引入 `symmetryPlane` 边界的时候）。每个算例中都有对网格生成方法的简明介绍（比如 `blockMesh`）。当然，用户可以直接查看网格，因为 OpenFOAM 当中每一个算例都含有 `polyMesh` 文件夹，并且在这个文件夹中有描述网格的数据文件。

这些例子都位于 OpenFOAM 安装路径下一个名为 `tutorials` 的子文件夹下。他们根据相应的求解器被分配到各个子文件夹中。比如说，每一个 `icoFoam` 对应的例子都被存储于一个名叫 `icoFoam` 的子文件夹中。在运行算例之前，我们建议用户将算例复制到其他文件夹之中。我们推荐用户将 OpenFOAM 算例存储于 `$FOAM_RUN` 文件夹中。如果这个文件夹还没有被创建，用户可以用如下命令创建此文件夹：

```
mkdir -p $FOAM_RUN
```

教程中的算例可以用如下命令复制到此文件夹中：

```
cp -r $FOAM_TUTORIALS/* $FOAM_RUN
```

### 3.1 圆柱绕流

在这个例子当中，我们将利用 `potentialFoam` 来研究圆柱势流（potential flow）。这个例子将会着重介绍 OpenFOAM 如何处理如下问题：

- 非正交网格；
- 对 OpenFOAM 当中的问题生成解析解；

### 3.1.1 问题阐述

我们将这个圆柱绕流问题描述如下：

**计算域：** 计算域是二维的，由一个正方形和一个与其同心的圆组成，如图 3.1 所示

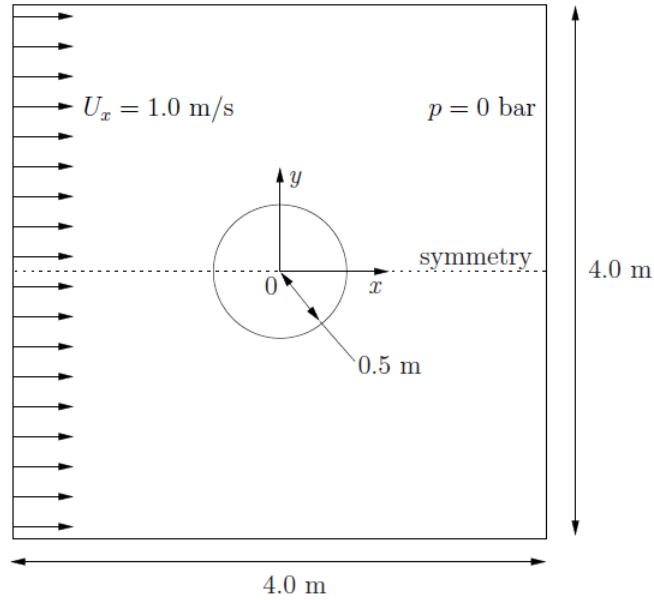


图 3.1 圆柱绕流几何模型示意图

**控制方程：**

- 不可压缩流体的质量守恒方程：

$$\nabla \cdot \mathbf{U} = 0 \quad (3.1)$$

- 不可压缩无旋场的稳态压力方程：

$$\nabla^2 p = 0 \quad (3.2)$$

**边界条件：**

- 入口（左边界）为固定速度入口  $\mathbf{U} = (1, 0, 0)$  m/s；
- 出口（右边界）为固定压力出口  $p = 0$  pa；
- 下边界为无滑移壁面；
- 上边界为对称面边界；

**初始条件：**  $\mathbf{U} = 0$  m/s 且  $p = 0$  pa。虽然对于稳态问题来说，并不需要初始条件，但是对于 OpenFOAM 算例来说，无论求解问题是否稳态都必须给定初始条件；

**求解器名称：** potentialFoam，其为求解势流的求解器，假设流场不可压缩、无旋、无黏、稳态、并且忽略重力；

**算例名称：** cylinder，这个算例位于 \$FOAM\_TUTORIALS/potentialFoam 文件夹中；

### 3.1.2 potentialFoam 求解器的注意事项

potentialFoam 求解器对于 OpenFOAM 正确性的验证有着十分重要的作用，因为势流中的假设使得许多几何模型比较简单的算例都具有解析解。对于这个圆柱绕流这个算例，我们可以用已有的解析解来对照我们的数值解。potentialFoam 还可以对其他问题提供符合质量守恒方程的初始场<sup>9</sup>。当处理类似问题的时候，此种方法可以避免由于初始场不稳定所带来的计算不稳定性。总之，potentialFoam 可以用来从用户提供的不符合质量守恒的初始场生成符合质量守恒的流场。

### 3.1.3 网格的生成

在《OpenFOAM 用户指南》中，我们已经阐述了如何用 blockMesh 生成网格。在这个算例中，整个计算域由十个 block 构成，如图 3.2 所示。需要注意的是，在 OpenFOAM 之中，所有网格都是按照三维网格处理的。如果我们想要解决二维问题，我们必须在第三个维度上加上一个网格的厚度（虽然我们并没有在这第三个维度上解控制方程）。

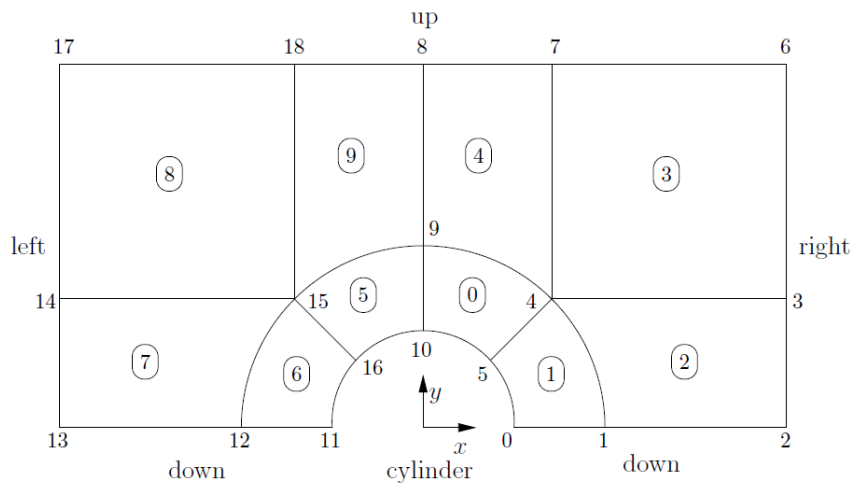


图 3.2 圆柱绕流中的 block

在图 3.2 中，我们只显示了几何图形的背面，即  $z = -0.5$  平面，在这个平面上，只有 0 到 18 号节点，而另外 19 个节点在几何图形的正面， $z = 0.5$  平面。正面 19 个节点的排序和背面相同，网格节点和 block 的信息被储存在 blockMeshDict 文件夹当中，如下所示：

```

1  /*-----C++-----*/
2  |=====|
3  |\ \ / / | Field      | OpenFOAM:   The Open Source CFD Toolbox
4  | \ \ / / | Operation | Version:   2.3.1
5  |  \ \ /  | And       | Web:       www.OpenFOAM.org
6  |   \ \ /   | Manipulation |
7  /*-----*/
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     object        blockMeshDict;

```

<sup>9</sup> 在算例运行之前首先运行势流求解器可以大大加快收敛且增加求解稳定性

---

```

14 }
15 // ****
16
17 convertToMeters 1;
18
19 vertices #codeStream
20 {
21     codeInclude
22     #{
23         #include "pointField.H"
24     #};
25
26     code
27     #{
28         pointField points(19);
29         points[0] = point(0.5, 0, -0.5);
30         points[1] = point(1, 0, -0.5);
31         points[2] = point(2, 0, -0.5);
32         points[3] = point(2, 0.707107, -0.5);
33         points[4] = point(0.707107, 0.707107, -0.5);
34         points[5] = point(0.353553, 0.353553, -0.5);
35         points[6] = point(2, 2, -0.5);
36         points[7] = point(0.707107, 2, -0.5);
37         points[8] = point(0, 2, -0.5);
38         points[9] = point(0, 1, -0.5);
39         points[10] = point(0, 0.5, -0.5);
40         points[11] = point(-0.5, 0, -0.5);
41         points[12] = point(-1, 0, -0.5);
42         points[13] = point(-2, 0, -0.5);
43         points[14] = point(-2, 0.707107, -0.5);
44         points[15] = point(-0.707107, 0.707107, -0.5);
45         points[16] = point(-0.353553, 0.353553, -0.5);
46         points[17] = point(-2, 2, -0.5);
47         points[18] = point(-0.707107, 2, -0.5);
48
49         // Duplicate z points
50         label sz = points.size();
51         points.setSize(2*sz);
52         for (label i = 0; i < sz; i++)
53         {
54             const point& pt = points[i];
55             points[i+sz] = point(pt.x(), pt.y(), -pt.z());
56         }
57
58         os << points;
59     #};
60 };
61
62
63 blocks
64 (
65     hex (5 4 9 10 24 23 28 29) (10 10 1) simpleGrading (1 1 1)
66     hex (0 1 4 5 19 20 23 24) (10 10 1) simpleGrading (1 1 1)
67     hex (1 2 3 4 20 21 22 23) (20 10 1) simpleGrading (1 1 1)
68     hex (4 3 6 7 23 22 25 26) (20 20 1) simpleGrading (1 1 1)
69     hex (9 4 7 8 28 23 26 27) (10 20 1) simpleGrading (1 1 1)
70     hex (15 16 10 9 34 35 29 28) (10 10 1) simpleGrading (1 1 1)
71     hex (12 11 16 15 31 30 35 34) (10 10 1) simpleGrading (1 1 1)
72     hex (13 12 15 14 32 31 34 33) (20 10 1) simpleGrading (1 1 1)
73     hex (14 15 18 17 33 34 37 36) (20 20 1) simpleGrading (1 1 1)
74     hex (15 9 8 18 34 28 27 37) (10 20 1) simpleGrading (1 1 1)
75 );
76
77 edges
78 (
79     arc 0 5 (0.469846 0.17101 -0.5)
80     arc 5 10 (0.17101 0.469846 -0.5)
81     arc 1 4 (0.939693 0.34202 -0.5)
82     arc 4 9 (0.34202 0.939693 -0.5)
83     arc 19 24 (0.469846 0.17101 0.5)
84     arc 24 29 (0.17101 0.469846 0.5)
85     arc 20 23 (0.939693 0.34202 0.5)
86     arc 23 28 (0.34202 0.939693 0.5)
87     arc 11 16 (-0.469846 0.17101 -0.5)
88     arc 16 10 (-0.17101 0.469846 -0.5)
89     arc 12 15 (-0.939693 0.34202 -0.5)
90     arc 15 9 (-0.34202 0.939693 -0.5)
91     arc 30 35 (-0.469846 0.17101 0.5)
92     arc 35 29 (-0.17101 0.469846 0.5)
93     arc 31 34 (-0.939693 0.34202 0.5)

```

```

94         arc 34 28 (-0.34202 0.939693 0.5)
95     );
96
97     boundary
98     (
99         down
100        {
101            type    symmetryPlane;
102            faces
103            (
104                (0 1 20 19)
105                (1 2 21 20)
106                (12 11 30 31)
107                (13 12 31 32)
108            );
109        }
110        right
111        {
112            type    patch;
113            faces
114            (
115                (2 3 22 21)
116                (3 6 25 22)
117            );
118        }
119        up
120        {
121            type    symmetryPlane;
122            faces
123            (
124                (7 8 27 26)
125                (6 7 26 25)
126                (8 18 37 27)
127                (18 17 36 37)
128            );
129        }
130        left
131        {
132            type    patch;
133            faces
134            (
135                (14 13 32 33)
136                (17 14 33 36)
137            );
138        }
139        cylinder
140        {
141            type    symmetry;
142            faces
143            (
144                (10 5 24 29)
145                (5 0 19 24)
146                (16 10 29 35)
147                (11 16 35 30)
148            );
149        }
150    );
151
152    mergePatchPairs
153    (
154    );
155
156    // *****

```

### 3.1.4 边界条件和初始条件

下面我们根据图 3.1 所示使用 FoamX<sup>10</sup>或者手动设置边界条件。左边界应该是入口（Inlet），右边界应该是出口（Outlet），下边界和圆柱边界应该设置为 `symmetryPlane`。当我们选择上边界的时候应当遵循以下规则：所选择的边界条件应当允许我们用理论解与数值解进行比较，因此，我们假设计算域在  $y$  方向上无穷大。这个假设使得垂直于上边界的速度梯度特别

<sup>10</sup> 下文提及的所有 FoamX 均停止更新，可采用其他的 OpenFOAM GUI 处理

小。因此我们在上边界选择垂直于上边界速度梯度为 0 的边界条件，即 `symmetryPlane`。这个边界条件使得理论解和数值解之间的比较变得更加合理。

### 3.1.5 运行算例

在这个算例中，我们不需要设置流体的性质，因为整个流场是不可压缩无旋无黏的。在 `system` 子文件夹中，`controlDict` 文件夹记录着算例运行所需要的参数。需要注意的是，因为我们要处理的是稳态问题，所以这个算例只应当运行一个时间步：

```

1  /*----- C++ -----*\
2  |=====|
3  |\ \ / / Field      | OpenFOAM:   The Open Source CFD Toolbox
4  | \ \ / / O peration | Version:    2.3.1
5  |  \ \ / / A nd      | Web:         www.OpenFOAM.org
6  |   \ \ / / M anipulation |
7  \*-----*/
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     location      "system";
14     object        controlDict;
15 }
16 // ***** //
17
18 application      potentialFoam;
19
20 startFrom        startTime;
21
22 startTime        0;
23
24 stopAt           endTime;
25
26 endTime          1;
27
28 deltaT           1;
29
30 writeControl      timeStep;
31
32 writeInterval     1;
33
34 purgeWrite        0;
35
36 writeFormat       ascii;
37
38 writePrecision    6;
39
40 writeCompression  off;
41
42 timeFormat        general;
43
44 timePrecision     6;
45
46 runTimeModifiable true;
47
48 functions
49 {
50     difference
51     {
52         // Load the library containing the 'coded' functionObject
53         functionObjectLibs ("libutilityFunctionObjects.so");
54         type coded;
55         // Name of on-the-fly generated functionObject
56         redirectType error;
57         code
58         #{
59             // Lookup U
60             Info<< "Looking up field U\n" << endl;
61             const volVectorField& U = mesh().lookupObject<volVectorField>("U");
62
63             Info<< "Reading inlet velocity uInFX\n" << endl;

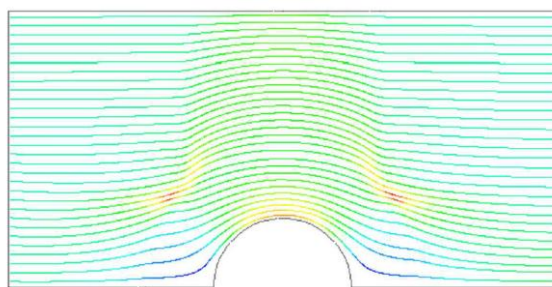
```

```

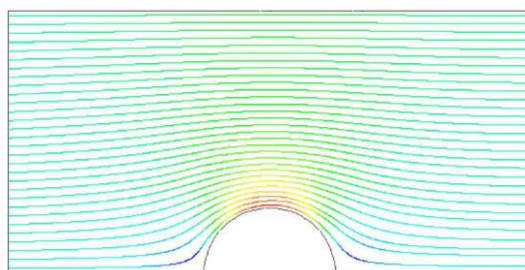
64
65     scalar ULeft = 0.0;
66     label leftI = mesh().boundaryMesh().findPatchID("left");
67     const fvPatchVectorField& fvp = U.boundaryField()[leftI];
68     if (fvp.size())
69     {
70         ULeft = fvp[0].x();
71     }
72     reduce(ULeft, maxOp<scalar>());
73
74     dimensionedScalar uInfX
75     (
76         "uInfX",
77         dimensionSet(0, 1, -1, 0, 0),
78         ULeft
79     );
80
81     Info << "U at inlet = " << uInfX.value() << " m/s" << endl;
82
83
84     scalar magCylinder = 0.0;
85     label cylI = mesh().boundaryMesh().findPatchID("cylinder");
86     const fvPatchVectorField& cylFvp = mesh().C().boundaryField()[cylI];
87     if (cylFvp.size())
88     {
89         magCylinder = mag(cylFvp[0]);
90     }
91     reduce(magCylinder, maxOp<scalar>());
92
93     dimensionedScalar radius
94     (
95         "radius",
96         dimensionSet(0, 1, 0, 0, 0),
97         magCylinder
98     );
99
100    Info << "Cylinder radius = " << radius.value() << " m" << endl;
101
102    volVectorField UA
103    (
104        IOobject
105        (
106            "UA",
107            mesh().time().timeName(),
108            U.mesh(),
109            IOobject::NO_READ,
110            IOobject::AUTO_WRITE
111        ),
112        U
113    );
114
115    Info<< "\nEvaluating analytical solution" << endl;
116
117    const volVectorField& centres = UA.mesh().C();
118    volScalarField magCentres(mag(centres));
119    volScalarField theta(acos((centres & vector(1,0,0))/magCentres));
120
121    volVectorField cs2theta
122    (
123        cos(2*theta)*vector(1,0,0)
124        + sin(2*theta)*vector(0,1,0)
125    );
126
127    UA = uInfX*(dimensionedVector(vector(1,0,0))
128    - pow((radius/magCentres),2)*cs2theta);
129
130    // Force writing of UA (since time has not changed)
131    UA.write();
132
133    volScalarField error("error", mag(U-UA)/mag(UA));
134
135    Info<<"Writing relative error in U to " << error.objectPath()
136    << endl;
137
138    error.write();
139    #};
140    }
141 }
142
143

```

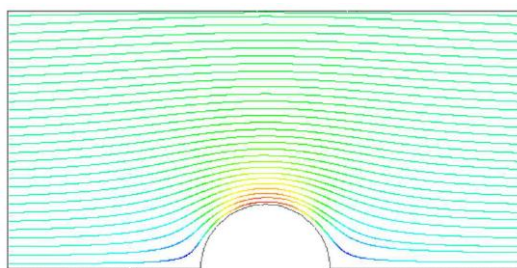
`potentialFoam`通过迭代来求解压力方程，并且可以对压力方程中的拉普拉斯项进行非正交修正（压力方程中的显式项会在正交修正迭代之后进行更新）。对压力方程的非正交修正迭代次数由`controlDict`之中的`nNonOrthogonalCorrectors`来决定。在第一个例子中，我们将`nNonOrthogonalCorrectors`设置为 0，也就是说，在求解过程中没有非正交修正迭代步（压力方程只被求解一次）。与之对应的数值解如图3.3（a）所示（ $t = 1$ ，当稳态求解结束之时）。我们期待数值解能够像解析解（如图3.3（c）所示）一样，即穿过整个计算域的流线是光滑的，但是在网格正交性不好的地方我们却能观察到明显的误差（比如在`block 0, 1, 3`的交界处）。如果我们将`nNonOrthogonalCorrectors`设置为3，再运行一次算例，其数值解如图3.3（b）所示。这个解的结果与理论解非常相近（流线非常光滑），因此并没有明显的由非正交性所引起的误差。



(a) 无非正交修正



(b) 有非正交修正



(c) 解析解

图 3.3 势流中的流线



## 3.2 稳态后向台阶湍流模拟

在这个算例中，我们将会进行稳态后向台阶湍流模拟。算例的原型来自 Pitz 和 Daily 的实验研究，我们将对算例所得到的数值解与实验值进行比较。这个算例会涉及如下特性：

- 利用 blockMesh 当中的网格非均匀分布功能生成网格；
- 稳态湍流；

### 3.2.1 问题阐述

稳态后向台阶湍流模拟定义如下：

**计算域：** 计算域是二维的，由一个窄的入口、一个后向台阶、一个在出口处收缩的喷管组成，如图 3.4 所示：

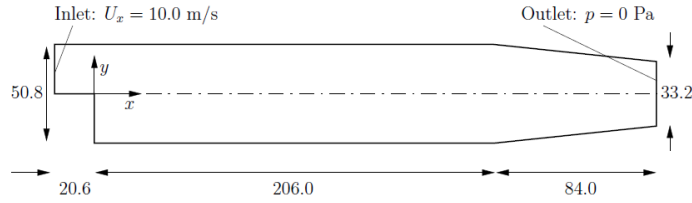


图 3.4 后向台阶的几何模型

**控制方程：**

- 不可压缩流体的质量守恒方程：

$$\nabla \cdot \mathbf{U} = 0 \quad (3.3)$$

- 稳态动量方程：

$$\nabla \cdot (\mathbf{UU}) + \nabla \cdot \mathbf{R} = -\nabla p \quad (3.4)$$

这里的  $p$  是动压， $\mathbf{R} = \nu_{eff} \nabla \mathbf{U}$  是带有有效动力粘度系数  $\nu_{eff}$  的粘性力项。 $\nu_{eff}$  是通过传递模型和湍流模型计算而来；

**初始条件：**  $\mathbf{U} = 0$  m/s 且  $p = 0$  pa。虽然对于稳态问题来说，并不需要初始条件，但是对于 OpenFOAM 算例来说，必须给定初始条件，无论求解问题是否稳态；

**边界条件：**

- 进口为固定速度入口， $\mathbf{U} = (10, 0, 0)$  m/s；
- 出口为固定压力出口  $p = 0$  pa；
- 其他边界都为非滑移边界；

```

1  /*-----*- C++ -*/
2  |=====|
3  |\\      /| Field      | OpenFOAM:      The Open Source CFD Toolbox
4  |  \\    /| Operation  | Version:      2.3.1
5  |   \\  /| And         | Web:        www.OpenFOAM.org
6  |    \\ /| Manipulation|
7  /*-----*-*/
8  FoamFile
9  {
10     version 2.0;
11     format      ascii;
12     class       dictionary;
13     object      blockMeshDict;
14 }
15 // *****
16
17 convertToMeters 0.001;
18
19 vertices
20 (
21     (-20.6 0 -0.5)
22     (-20.6 3 -0.5)
23     (-20.6 12.7 -0.5)
24     (-20.6 25.4 -0.5)
25     (0 -25.4 -0.5)

```

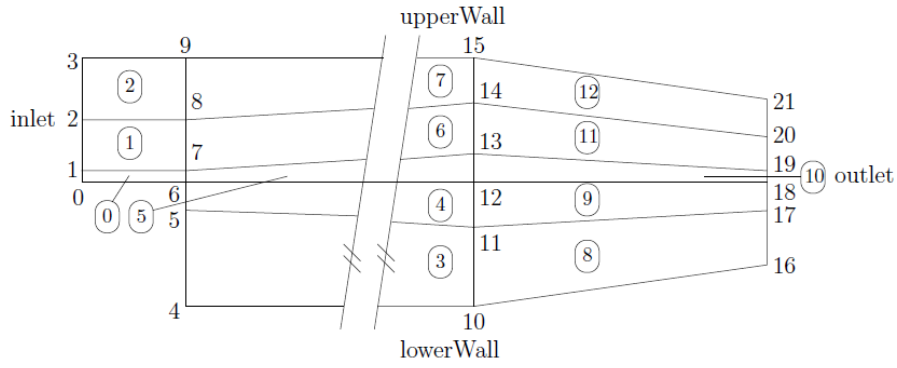


图 3.5 后向台阶问题中的 block

```

26      (0 -5 -0.5)
27      (0 0 -0.5)
28      (0 3 -0.5)
29      (0 12.7 -0.5)
30      (0 25.4 -0.5)
31      (206 -25.4 -0.5)
32      (206 -8.5 -0.5)
33      (206 0 -0.5)
34      (206 6.5 -0.5)
35      (206 17 -0.5)
36      (206 25.4 -0.5)
37      (290 -16.6 -0.5)
38      (290 -6.3 -0.5)
39      (290 0 -0.5)
40      (290 4.5 -0.5)
41      (290 11 -0.5)
42      (290 16.6 -0.5)
43      (-20.6 0 0.5)
44      (-20.6 3 0.5)
45      (-20.6 12.7 0.5)
46      (-20.6 25.4 0.5)
47      (0 -25.4 0.5)
48      (0 -5 0.5)
49      (0 0 0.5)
50      (0 3 0.5)
51      (0 12.7 0.5)
52      (0 25.4 0.5)
53      (206 -25.4 0.5)
54      (206 -8.5 0.5)
55      (206 0 0.5)
56      (206 6.5 0.5)
57      (206 17 0.5)
58      (206 25.4 0.5)
59      (290 -16.6 0.5)
60      (290 -6.3 0.5)
61      (290 0 0.5)
62      (290 4.5 0.5)
63      (290 11 0.5)
64      (290 16.6 0.5)
65 );
66
67 blocks
68 (
69     hex (0 6 7 1 22 28 29 23) (18 7 1) simpleGrading (0.5 1.8 1)
70     hex (1 7 8 2 23 29 30 24) (18 10 1) simpleGrading (0.5 4 1)
71     hex (2 8 9 3 24 30 31 25) (18 13 1) simpleGrading (0.5 0.25 1)
72     hex (4 10 11 5 26 32 33 27) (180 18 1) simpleGrading (4 1 1)
73     hex (5 11 12 6 27 33 34 28) (180 9 1) edgeGrading (4 4 4 4 0.5 1 1 0.5 1 1 1 1)
74     hex (6 12 13 7 28 34 35 29) (180 7 1) edgeGrading (4 4 4 4 1.8 1 1 1.8 1 1 1 1)
75     hex (7 13 14 8 29 35 36 30) (180 10 1) edgeGrading (4 4 4 4 4 1 1 4 1 1 1 1)
76     hex (8 14 15 9 30 36 37 31) (180 13 1) simpleGrading (4 0.25 1)
77     hex (10 16 17 11 32 38 39 33) (25 18 1) simpleGrading (2.5 1 1)
78     hex (11 17 18 12 33 39 40 34) (25 9 1) simpleGrading (2.5 1 1)
79     hex (12 18 19 13 34 40 41 35) (25 7 1) simpleGrading (2.5 1 1)
80     hex (13 19 20 14 35 41 42 36) (25 10 1) simpleGrading (2.5 1 1)
81     hex (14 20 21 15 36 42 43 37) (25 13 1) simpleGrading (2.5 0.25 1)
82 );
83
84 edges
85 (

```

```

86 );
87
88 boundary
89 (
90     inlet
91     {
92         type    patch;
93         faces
94         (
95             (0 22 23 1)
96             (1 23 24 2)
97             (2 24 25 3)
98         );
99     }
100    outlet
101    {
102        type    patch;
103        faces
104        (
105            (16 17 39 38)
106            (17 18 40 39)
107            (18 19 41 40)
108            (19 20 42 41)
109            (20 21 43 42)
110        );
111    }
112    upperWall
113    {
114        type    wall;
115        faces
116        (
117            (3 25 31 9)
118            (9 31 37 15)
119            (15 37 43 21)
120        );
121    }
122    lowerWall
123    {
124        type    wall;
125        faces
126        (
127            (0 6 28 22)
128            (6 5 27 28)
129            (5 4 26 27)
130            (4 10 32 26)
131            (10 16 38 32)
132        );
133    }
134    frontAndBack
135    {
136        type    empty;
137        faces
138        (
139            (22 28 29 23)
140            (23 29 30 24)
141            (24 30 31 25)
142            (26 32 33 27)
143            (27 33 34 28)
144            (28 34 35 29)
145            (29 35 36 30)
146            (30 36 37 31)
147            (32 38 39 33)
148            (33 39 40 34)
149            (34 40 41 35)
150            (35 41 42 36)
151            (36 42 43 37)
152            (0 1 7 6)
153            (1 2 8 7)
154            (2 3 9 8)
155            (4 5 11 10)
156            (5 6 12 11)
157            (6 7 13 12)
158            (7 8 14 13)
159            (8 9 15 14)
160            (10 11 17 16)
161            (11 12 18 17)
162            (12 13 19 18)
163            (13 14 20 19)
164            (14 15 21 20)
165        );

```

```

166     }
167 );
168
169 mergePatchPairs
170 (
171 );
172
173 // ***** //

```

这个算例的一大特点,是它大量的使用了《OpenFOAM用户指南》中 5.3.1 所介绍的 **blockMesh** 中的网格非均匀功能。用户可以看到,在 **block 4、5、6** 中,每个边均使用了非均匀分布(每个膨胀率都对应着 **block** 其中的一个边)。前四个膨胀率对应与  $x_1$  轴平行的四个边,中间四个膨胀率对应与  $x_2$  轴平行的四个边,后四个膨胀率对应与  $x_3$  轴平行的四个边。在 **block 4、5、6** 中,膨胀率在  $x_1$  和  $x_3$  方向上都是相同的,但是在  $x_2$  方向( $y$  方向)却是不同的。如果我们参考《OpenFOAM 用户指南》5.3.1 中对 **block** 的定义,我们可以发现 **block 4、5、6** 中的左边和右边的细化是不一样的。这样做的目的是为了在最关键的地方(台阶的尖角处)生成细网格。

网格可以在终端由 **bockMesh** 生成,也可以在 **FoamX** 之中生成,查看网格的方法再次不再赘述。

### 3.2.3 边界条件和初始场

与算例相关的参数文件可以在 **FoamX** 之中浏览或者修改(当然也可以用手动修改)。在这个算例中,我们需要为如下参数设置边界条件和初始条件:速度  $\mathbf{U}$ , 压力  $p$ , 湍流动能  $k$ , 和耗散率  $\varepsilon$ 。边界条件的设置可以通过在 **FoamX** 之中设置 **patch** 的类型来实现:上边界和下边界被设置为 **wall**, 左边界被设置为 **Inlet**, 右边界被设置为 **Outlet**。这样的边界条件组合需要我们在入口处设定  $\mathbf{U}$ 、 $k$ 、 $\varepsilon$  的类型为 **fixedValue**。 $\mathbf{U}$  的具体值已经在问题阐述中给出。 $k$  和  $\varepsilon$  的值应该利用和《OpenFOAM 用户指南》中 2.1.8.1 节一样的方法计算出来。我们假设入口湍流是各向同性,而且入口处的速度波动为 5%有:

$$U'_x = U'_y = U'_z = \frac{5}{100} 10 = 0.5 \text{ m/s} \quad (3.5)$$

且

$$k = 1.5(0.5)^2 = 0.375 \text{ m}^2/\text{s}^2 \quad (3.6)$$

如果我们假定湍流尺度为入口边界长度的十分之一有:

$$\varepsilon = \frac{C_\mu^{0.75} k^{1.5}}{l} = \frac{0.09^{0.75} 0.375^{1.5}}{0.1 \times 25.4 \times 10^{-3}} = 14855 \text{ m}^2/\text{s}^2 \quad (3.7)$$

在出口处,我们只需要将  $p$  设置为 0 pa。

### 3.2.4 求解器的控制参数

**fvSchemes** 的设置如下: **timeScheme** 设置为 **SteadyState**; **gradScheme** 和 **laplacianScheme** 设置为默认值 **Gauss**; 为了确保有界性, **divScheme** 应该设置为 **UD**。

我们需要特殊注意 **fvTolerances** 的设置。虽然顶层的 **simpleFoam** 代码只包含了压力和速度的方程,但是在湍流模型中,我们还要求解  $k$ ,  $\varepsilon$  和  $R$  的方程,所以,我们需要为 5 个方程设置残差(**tolerance**)。除了将压力的 **solverTolerance** 设置为  $10^{-6}$  且 **solverRelativeTolerance** 设置为 0.01 之外,对于其他变量,我们将这两个参数分别设置为  $10^{-5}$  和 0.1。由于这个算例是一个稳态问题,所以我们需要设置松弛因子(**relaxationFactor**)。对于压力,为了避免计算的不稳定,松弛因子被设置为 0.3。对于其他参数( $\mathbf{U}$ ,  $k$ ,  $\varepsilon$ ,  $R$ ),松弛因子可以设置为 0.7。

最后,在 **controlDict** 之中,我们将时间步长设置为 1(因为计算的稳态的),这里的时间

步长只用于稳态计算中的迭代次数进行计数。这个算例需要至少一千次迭代才能收敛,所以,我们将 `endTime` 设置为 1000。我们还需要保证存储足够的时间步且不会太多,这样可以设置为 50 步一写,以保证用户的硬盘空间不会在算例运行过程中被数据填满。

### 3.2.5 运行算例和后处理

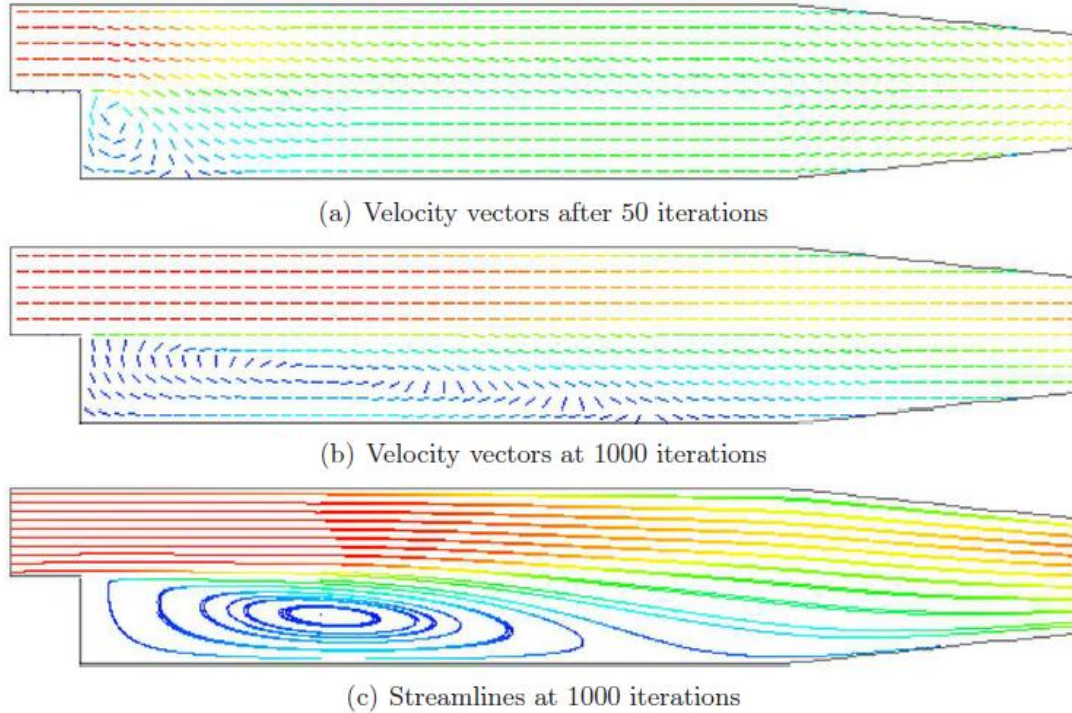


图 3.6 后面台阶问题中的漩涡发展

接下来我们运行程序并且对运行结果进行后处理。在几步迭代之后,比如 50 步,在台阶的尖角下方会出现一个漩涡,但是漩涡在  $x$  方向上长度很短,如图 3.6 (a) 所示。在 1000 个时间步之后,漩涡沿着  $x$  方向从台阶处一直延伸到出口处,此时系统已经达到稳态,并且漩涡已经充分发展,如图 3.6 (b-c) 所示。

## 3.3 前向台阶的超音速绕流

在这个算例中,我们将会探究前向台阶的超音速绕流。在这个算例中,入口速度马赫数为 3,在入口附近有一个台阶,在台阶附近会产生激波。

这个算例主要涉及 OpenFOAM 的如下特性:

- 超音速问题的求解

### 3.3.1 问题阐述

**计算域:** 计算域是二维的,并且由一个入口区域和一个前向台阶组成。台阶的高度占入口长

度的 20%，如图 3.7 所示：

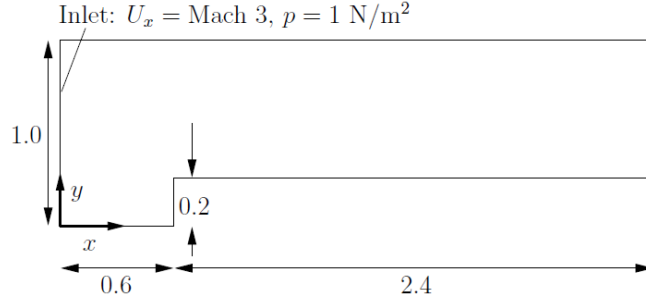


图 3.7 前向台阶的几何模型

**控制方程:**

- 质量守恒方程：

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \quad (3.8)$$

- 理想气体状态方程：

$$p = \rho RT \quad (3.9)$$

- 牛顿流体的动量守恒方程：

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) - \nabla \cdot \mu \nabla \mathbf{U} = -\nabla p \quad (3.10)$$

- 能量方程（忽略粘性项）， $e = C_v T$ ，和傅里叶定律， $\mathbf{q} = -k \nabla T$ ：

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho \mathbf{U} e) - \nabla \cdot \left( \frac{k}{C_v} \nabla e \right) = -p \nabla \cdot \mathbf{U} \quad (3.11)$$

**初始条件:**  $U = 0 \text{ m/s}$ ,  $p = 1 \text{ pa}$ ,  $T = 1 \text{ k}$ ;

**边界条件:**

- 入口设置为固定值边界条件 `fixedValue`， $U = 3 \text{ m/s} = \text{Mach } 3$ ,  $p = 1 \text{ pa}$ ,  $T = 1 \text{ k}$ ;
- 出口对速度，压力，温度（ $U$ ,  $p$ ,  $T$ ）设置为法相梯度为零 `zeroGradient`;
- 下边界被设置为无滑移绝热边界;
- 上边界被设置为 `symmetryPlane`;

**物性:**

- 空气的动力粘度系数为  $\mu = 18.1 \mu\text{Pas}$ ;

**热力学性质:**

- 定压比热容  $C_v = 1.78571 \text{ J/kgK}$ ;
- 气体常数  $R = 0.714286 \text{ J/kgK}$ ;

- 热导率  $k = 32.3 \mu\text{W/mK}$  ;

**算例名称:** forwardStep, 其位于\$FOAM\_TUTORIALS/sonicFoam 文件夹中;

**求解器名称:** sonicFoam: 为可压缩流体的超音速和跨音速问题设计的求解器;

在本算例中, 音速被设计为  $1\text{m/s}$  ( $c = \sqrt{\gamma RT} = 1\text{m/s}$ )。这样做的结果就是速度的大小直接等于马赫数。音速的大小可以利用如下关系计算。对于理想气体,  $C_p - C_v = R$ :

$$\gamma = \frac{C_p}{C_v} = \frac{R}{C_v} + 1 \quad (3.12)$$

### 3.3.2 网格生成

本算例中所应用的网格比较简单, 用大小相同的长方体网格生成即可。这些长方体网格在  $x$  方向长  $0.06\text{m}$ ,  $y$  方向长  $0.05\text{m}$ 。整个计算域可以被分为三个 block, 一个在台阶上沿下方, 两个在台阶上沿上方 (分列于台阶两侧)。关于网格的信息被储存在如下所示的 blockMeshDict 文件之中:

```

1  /*----- C++ -----*\
2  |=====|
3  | \ \ / Field | OpenFOAM: The Open Source CFD Toolbox |
4  | \ \ / Operation | Version: 2.3.1 |
5  | \ \ / A nd | Web: www.OpenFOAM.org |
6  | \ \ / M anipulation | |
7  \*-----*/
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     object        blockMeshDict;
14  }
15  // *****
16
17  convertToMeters 1;
18
19  vertices
20  (
21      (0 0 -0.05)
22      (0.6 0 -0.05)
23      (0 0.2 -0.05)
24      (0.6 0.2 -0.05)
25      (3 0.2 -0.05)
26      (0 1 -0.05)
27      (0.6 1 -0.05)
28      (3 1 -0.05)
29      (0 0 0.05)
30      (0.6 0 0.05)
31      (0 0.2 0.05)
32      (0.6 0.2 0.05)
33      (3 0.2 0.05)
34      (0 1 0.05)
35      (0.6 1 0.05)
36      (3 1 0.05)
37  );
38
39  blocks
40  (
41      hex (0 1 3 2 8 9 11 10) (25 10 1) simpleGrading (1 1 1)
42      hex (2 3 6 5 10 11 14 13) (25 40 1) simpleGrading (1 1 1)
43      hex (3 4 7 6 11 12 15 14) (100 40 1) simpleGrading (1 1 1)
44  );
45
46  edges
47  (

```



```

48 );
49
50 boundary
51 (
52     inlet
53     {
54         type    patch;
55         faces
56         (
57             (0 8 10 2)
58             (2 10 13 5)
59         );
60     }
61     outlet
62     {
63         type    patch;
64         faces
65         (
66             (4 7 15 12)
67         );
68     }
69     bottom
70     {
71         type    symmetryPlane;
72         faces
73         (
74             (0 1 9 8)
75         );
76     }
77     top
78     {
79         type    symmetryPlane;
80         faces
81         (
82             (5 13 14 6)
83             (6 14 15 7)
84         );
85     }
86     obstacle
87     {
88         type    patch;
89         faces
90         (
91             (1 3 11 9)
92             (3 4 12 11)
93         );
94     }
95 );
96
97 mergePatchPairs
98 (
99 );
100
101 // ***** //

```

### 3.3.3 运行算例

本算例将会在 5s 之后达到稳态。10 秒钟时的压力分布如图 3.8 所示。计算结果显示出了压力的明显不连续性，也就是激波。我们可以看到，激波由从台阶下沿的前方开始发展。

### 3.3.4 练习

用户可以检验入口速度的升高对于解的影响。

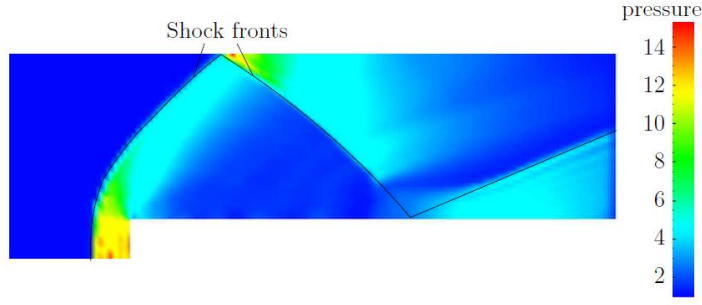


图 3.8 前向台阶问题中的激波

### 3.4 高压水箱放水问题

在这个算例中，我们将会探究一个装满水的高压水箱放水问题。这个问题的一大特点是压力波的传播，因此，这个算例必须要考虑流体的可压缩性。

这个算例涉及到了 OpenFOAM 的如下特性：

- 网格加密；
- 流体中的压力波；

#### 3.4.1 问题阐述

**计算域：** 计算域是二维的，由一个水箱和一小截泄流管组成，如图 3.9 所示：

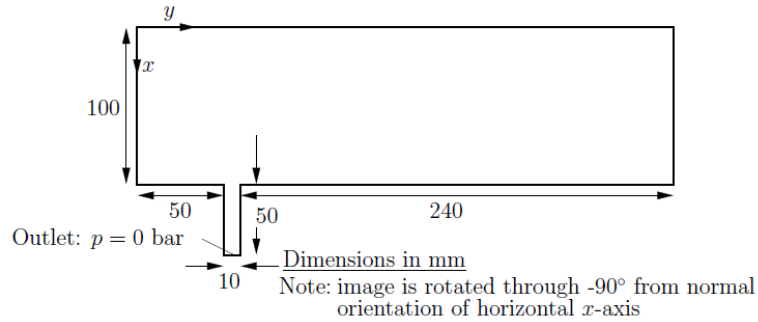


图 3.9 带有泄流管道的水箱示意图

**控制方程：** 在这个问题中，为了能够准确捕捉以有限速度传播的压力波，我们需要考虑流体的压缩性 $\psi$ 。我们将利用正压关系（barotropic relationship）来将压力、密度和压缩性 $\psi$ 联系在一起：

- 质量守恒方程：

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \quad (3.13)$$

- 正压关系:

$$\frac{\partial \rho}{\partial p} = \frac{\rho}{K} = \psi \quad (3.14)$$

K是体积模量;

- 公式 3.14 被线性化为:

$$\rho \approx \rho_0 + \psi(p - p_0) \quad (3.15)$$

$\rho_0$ 和 $p_0$ 分别为参考密度和参考压力,  $\rho(p_0) = \rho_0$ ;

- 牛顿流体的动量守恒方程:

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) - \nabla \cdot \mu \nabla \mathbf{U} = -\nabla p \quad (3.16)$$

**边界条件:** 使用 FoamX, 将边界条件设置如下:

- outerWall 设置为 wall;
- axis 设置为 symmetryPlane;
- outlet 设置为 pressureOutlet, 出口压力位稳态  $p=0$  bar;
- front 和 back 设置为 empty;

**初始条件:**  $U = 0\text{m/s}$ ,  $p = 100$  bar;

**物性:**

- 水的动力粘度系数为  $\mu = 1.0\text{mPas}$ ;

**热力学性质:**

- 水的密度:  $\rho = 1000\text{kg/m}^3$ ;
- 参考压力:  $p_0 = 1\text{ bar}$ ;
- 水的压缩性:  $\psi = 4.54 \times 10^{-7}\text{s}^2/\text{m}^2$ ;

**求解器名称:** sonicLiquidFoam, 用来求解可压缩超音速层流问题;

**算例名称:** decompressionTank, 这个算例位于\$FOAM\_TUTORIALS/sonicLiquidFoam 文件夹中;

### 3.4.2 网格的生成

下面为其 blockMeshDict 字典文件:

```

1  /*----- C++ -----*/
2  |=====|
3  |\ \      /  F i e l d      | OpenFOAM:   The Open Source CFD Toolbox |
4  | \ \      /  O p e r a t i o n | Version:   2.3.1 |
5  |  \ \      /  A n d              | Web:       www.OpenFOAM.org  |
6  |   \ \      /  M a n i p u l a t i o n |

```

```

7  \*-----*/
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class          dictionary;
13     object         blockMeshDict;
14  }
15  // *****
16
17  convertToMeters 0.1;
18
19  vertices
20  (
21      (0 0 -0.1)
22      (1 0 -0.1)
23      (0 0.5 -0.1)
24      (1 0.5 -0.1)
25      (1.5 0.5 -0.1)
26      (0 0.6 -0.1)
27      (1 0.6 -0.1)
28      (1.5 0.6 -0.1)
29      (0 3 -0.1)
30      (1 3 -0.1)
31      (0 0 0.1)
32      (1 0 0.1)
33      (0 0.5 0.1)
34      (1 0.5 0.1)
35      (1.5 0.5 0.1)
36      (0 0.6 0.1)
37      (1 0.6 0.1)
38      (1.5 0.6 0.1)
39      (0 3 0.1)
40      (1 3 0.1)
41  );
42
43  blocks
44  (
45      hex (0 1 3 2 10 11 13 12) (30 20 1) simpleGrading (1 1 1)
46      hex (2 3 6 5 12 13 16 15) (30 5 1) simpleGrading (1 1 1)
47      hex (3 4 7 6 13 14 17 16) (25 5 1) simpleGrading (1 1 1)
48      hex (5 6 9 8 15 16 19 18) (30 95 1) simpleGrading (1 1 1)
49  );
50
51  edges
52  (
53  );
54
55  boundary
56  (
57      outerWall
58      {
59          type    wall;
60          faces
61          (
62              (0 1 11 10)
63              (1 3 13 11)
64              (3 4 14 13)
65              (7 6 16 17)
66              (6 9 19 16)
67              (9 8 18 19)
68          );
69      }
70      axis
71      {
72          type    symmetryPlane;
73          faces
74          (
75              (0 10 12 2)
76              (2 12 15 5)
77              (5 15 18 8)
78          );
79      }
80      nozzle
81      {
82          type    patch;
83          faces
84          (
85              (4 7 17 14)
86          );

```

```

87     }
88     back
89     {
90         type    empty;
91         faces
92         (
93             (0 2 3 1)
94             (2 5 6 3)
95             (3 6 7 4)
96             (5 8 9 6)
97         );
98     }
99     front
100    {
101        type    empty;
102        faces
103        (
104            (10 11 13 12)
105            (12 13 16 15)
106            (13 14 17 16)
107            (15 16 19 18)
108        );
109    }
110 );
111
112 mergePatchPairs
113 (
114 );
115
116 // ****

```

为了提高计算的准确性，我们将参考压力设置为 1 bar。需要注意的是，初始场和边界条件中的压力都应该被参考压力值抵消一部分。

### 3.4.3 准备运行算例

在我们设置 `controlDict` 的运行参数之前，我们应该考虑一下本算例的特征速度。在本算例中，流体的速度是很小的，但是压力波却在水中以声速传播。水中的声速计算如下：

$$c = \sqrt{\frac{1}{\psi}} = \sqrt{\frac{1}{4.54 \times 10^{-7}}} = 1483.2 \text{ m/s} \quad (3.17)$$

对于上述网格，特征网格长度越为 2mm(请注意在 `blockMeshDict` 之中我们把 `convertToMeters` 被设置为 0.1)。

$$\text{Co} = \frac{U \Delta t}{\Delta x} \quad (3.18)$$

本算例合理的时间步应该为  $\Delta t = 5 \times 10^{-7} \text{ s}$ ，在此时间步长下，取水中的声速计算库郎数为 0.35。需要注意的是，因为我们想要捕捉的是压力波在水中的传播，所以程序在运行时所输出的 Co 数可能会比 0.35 小两个数量级。`controlDict` 之中的设置如下：

```

1  /*-----C++-----*\
2  |=====|
3  |\ \ /  | Field      | OpenFOAM:   The Open Source CFD Toolbox |
4  |\ \ /  | Operation  | Version:    2.3.1                      |
5  |\ \ /  | A nd       | Web:        www.OpenFOAM.org           |
6  |\ \ /  | M anipulation |
7  \-----*/
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     location      "system";
14     object        controlDict;
15 }
16 // ****
17
18 application      sonicLiquidFoam;

```

```

19
20 startFrom      startTime;
21
22 startTime      0;
23
24 stopAt         endTime;
25
26 endTime        0.0001;
27
28 deltaT         5e-07;
29
30 writeControl    timeStep;
31
32 writeInterval   20;
33
34 purgeWrite     0;
35
36 writeFormat     ascii;
37
38 writePrecision  6;
39
40 writeCompression off;
41
42 timeFormat      general;
43
44 timePrecision   6;
45
46 runTimeModifiable true;
47
48
49 // *****

```

### 3.4.4 运行算例

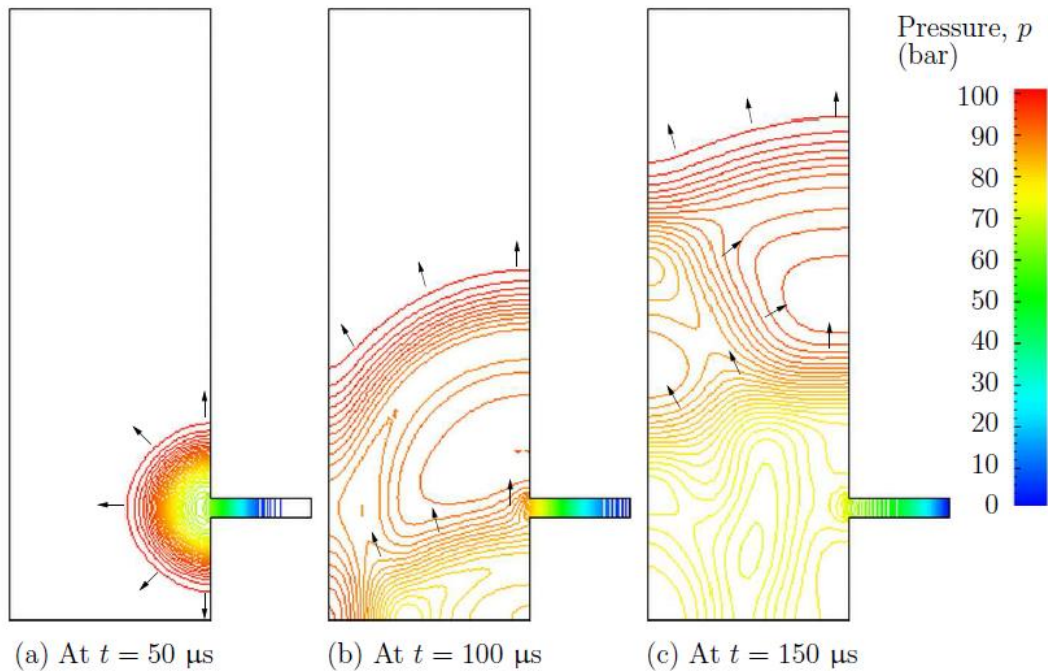


图 3.10 压力波的传播

用户可以运行算例，并且在 paraFoam 中观察运行结果。通过喷管流出的液体使得计算域中形成沿着喷管移动的压力波。当压力波到达喷管和水箱的连接处时，有的压力波通过连接处传入水箱中，有的被反射回喷管中。由于压力波在喷管中被来回反弹，通过连接处的压力波会膨胀至整个水箱。参见图 3.10 中的压力云图，这样做的目的是为了比从单纯的等值线更

清晰地观察压力波。

如果我们让算例运行足够长的时间，让压力波充分的反射回喷管，我们可以清楚地再在喷管中观察到绝对负压。这个现象非常合理且有物理依据，因为流体能够承受张力（比如说负压）。然而，在实际情况下，由于液体的非纯净性，液体中溶解的气体对流体流动会产生一定的空化效果（由于负压产生的气泡，沸腾和气化的效果一样）。因此，在实际情况下，我们不会观察到压力下降到气化压力之下（至少在产生气泡之前）。

### 3.4.5 细化网格来获得更精确解

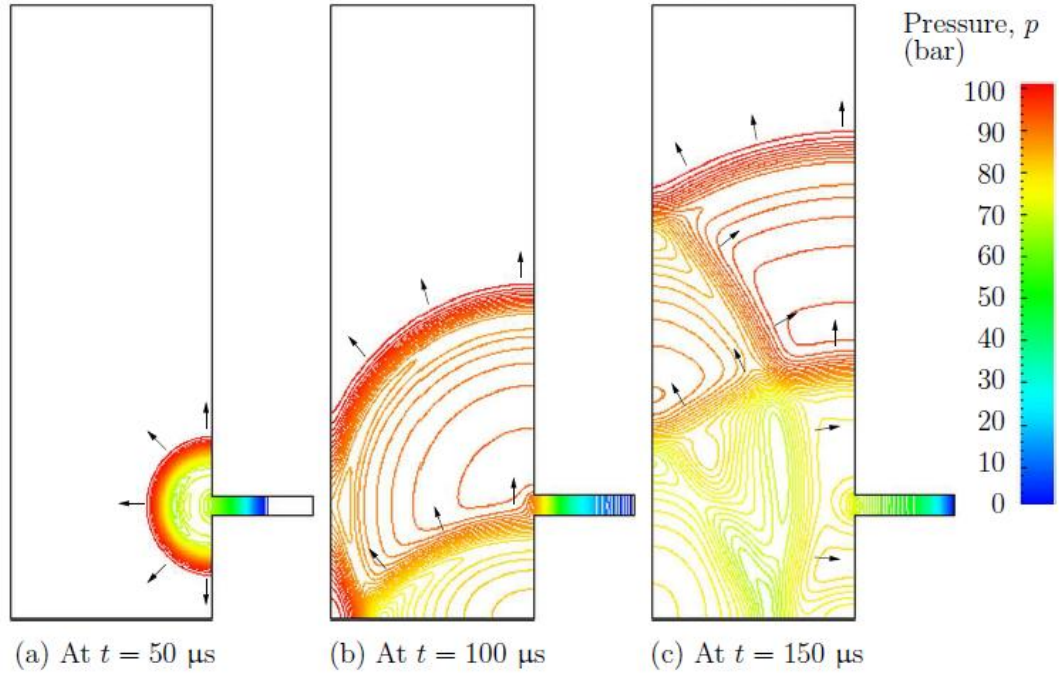


图 3.11 更加细化的网格中的压力波传播

通过观察压力场随时间的变化，我们可以看到压力波传入水箱中，并且在各个壁面之间进行反射。我们也可以清楚的看到压力波在某些网格区域有所聚集。现在，为了更好的捕捉压力波，我们对网格进行细化，并且减小时间步长。我们只需要将 `blockMeshDict` 里面的网格数乘以 4，比如 `block 0` 的网格数从 (30, 20, 1) 增加到 (120, 80, 1)。在文件目录下运行 `blockMesh`。除此之外，为了将 `Co` 数控制在 1 以内，我们也应该将时间步长缩小到  $\Delta t = 10^{-7} \text{s}$ 。如图 3.11 所示，加密之后的网格明显能够得到更精确的结果。

## 3.5 磁流体问题

在这个算例中，我们将会探究流过磁场的导电流体。这类问题被称为磁流体问题，我们用 `mhdFoam` 来解决类似问题。

### 3.5.1 问题阐述

这个算例被称为 **Hartmann** 问题，这个问题有解析解，我们可以用解析解来验证 **mhdFoam** 的正确性。这个问题的定义如下：

**计算域：** 计算域是二维的，由两个平行的平板组成，流体从两个平板之间流过，如图 3.12 所示：

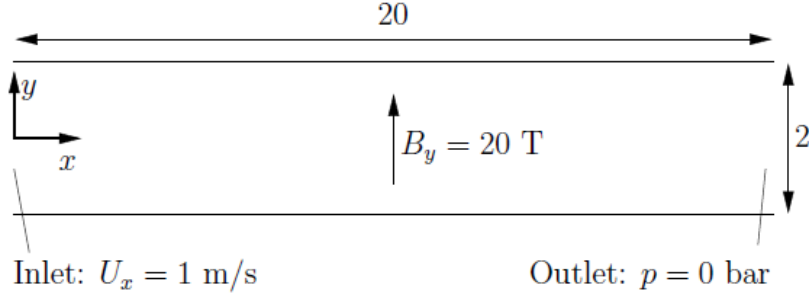


图 3.12 Hartmann 问题的几何模型

**控制方程：**

- 不可压缩流体的质量守恒方程：

$$\nabla \cdot \mathbf{U} = 0 \quad (3.19)$$

- 不可压缩流体的动量守恒方程：

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{U}\mathbf{U}) + \nabla \cdot (2\mathbf{B}\Gamma_{BU}\mathbf{B}) + \nabla \cdot (\nu\mathbf{U}) + \nabla(\Gamma_{BU}\mathbf{B}:\mathbf{B}) = -\nabla p \quad (3.20)$$

$\mathbf{B}$  为磁感应强度， $\Gamma_{BU} = (2\mu\rho)^{-1}$ ；

- 麦克斯韦方程组：

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (3.21)$$

$\mathbf{E}$  为电场强度；

$$\nabla \cdot \mathbf{B} = 0 \quad (3.22)$$

$$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} = \mathbf{J} \quad (3.23)$$

假设  $\frac{\partial \mathbf{D}}{\partial t} \ll \mathbf{J}$ ，这里  $\mathbf{H}$  为磁场强度， $\mathbf{J}$  为电流密度， $\mathbf{D}$  为电通量密度；

- 电流连续性：

$$\nabla \cdot \mathbf{J} = 0 \quad (3.24)$$



- 本构方程:

$$\mathbf{B} = \mu \mathbf{H} \quad (3.25)$$

- 欧姆定律:

$$\mathbf{J} = \sigma(\mathbf{E} + \mathbf{U} \times \mathbf{B}) \quad (3.26)$$

- 将方程 3.21, 3.23, 3.26 联立, 并且取旋度运算, 可以得到:

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{U} \mathbf{B}) - \nabla \cdot (\phi_B \mathbf{U}) - \nabla \cdot (\Gamma_{BU} \mathbf{B}) = 0 \quad (3.27)$$

**边界条件:**

- 入口条件被设置为固定值速度  $\mathbf{U} = (1, 0, 0)$  m/s;
- 出口条件被设置为稳态压力出口  $p = 0$  pa;
- 上边界被设置为壁面,  $\mathbf{B} = (0, 20, 0)$  T;
- 下边界被设置为壁面,  $\mathbf{B} = (0, 20, 0)$  T;
- 前后面边界被设置为 empty;

**初始条件:**  $\mathbf{U} = 0$  m/s,  $p = 100$  pa,  $\mathbf{B} = (0, 20, 0)$  T;

**物性:**

- 运动粘度系数:  $\nu = 1 \text{ m}^2/\text{s}$ ;
- 密度:  $\rho = 1 \text{ kg/m}^3$ ;
- 电导率:  $\sigma = 1(\Omega \text{ m})^{-1}$ ;
- 磁导率:  $\mu = 1 \text{ H/m}$ ;

**求解器名称:** mhdFoam: 用于求解不可压缩层流磁流体问题的求解器;

**算例名称:** hartmann, 这个算例位于 \$FOAM\_TUTORIALS/mhdFoam 文件夹中;

### 3.5.2 网格的生成

本算例的网格很简单, 在 x 方向有 100 个网格, 在 y 方向有 40 个网格。blockMeshDict 文件如下:

```

1  /*----- C++ -----*/
2  |=====|
3  |\\      /| Field      | OpenFOAM:   The Open Source CFD Toolbox |
4  | |\\    /| Operation  | Version:    2.3.1 |
5  | |\\    /| And        | Web:        www.OpenFOAM.org  |
6  | |\\    /| Manipulation| |
7  /*-----*/
8  FoamFile
9  {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     object        blockMeshDict;
14  }
15  // *****
16
17  convertToMeters 1;
```

```

18
19 vertices
20 (
21     (0 -1 0)
22     (20 -1 0)
23     (20 1 0)
24     (0 1 0)
25     (0 -1 0.1)
26     (20 -1 0.1)
27     (20 1 0.1)
28     (0 1 0.1)
29 );
30
31 blocks
32 (
33     hex (0 1 2 3 4 5 6 7) (100 40 1) simpleGrading (1 1 1)
34 );
35
36 edges
37 (
38 );
39
40 boundary
41 (
42     inlet
43     {
44         type patch;
45         faces
46         (
47             (0 4 7 3)
48         );
49     }
50     outlet
51     {
52         type patch;
53         faces
54         (
55             (2 6 5 1)
56         );
57     }
58     lowerWall
59     {
60         type patch;
61         faces
62         (
63             (1 5 4 0)
64         );
65     }
66     upperWall
67     {
68         type patch;
69         faces
70         (
71             (3 7 6 2)
72         );
73     }
74     frontAndBack
75     {
76         type empty;
77         faces
78         (
79             (0 3 2 1)
80             (4 5 6 7)
81         );
82     }
83 );
84
85 mergePatchPairs
86 (
87 );
88
89 // *****

```

### 3.5.3 运行算例

用户可以运行算例并且在 paraFoam 中查看运行结果。我们也可以用 Ucomponents 命令将向

量场  $U$  转化为  $U$  各个分量的标量场。磁流体问题与许多参数相关，这些参数构成了无量纲的 Hartmann 数。它表征电磁力和粘性力的比值：

$$M = BL \sqrt{\frac{\sigma}{\rho\nu}} \quad (3.28)$$

这里， $L$  代表问题的特征长度， $B_y = 20\text{T}$ ,  $M = 20$ 。在这个算例中，电磁力远大于粘性力。在计算结果中，流场在 2s 时达到稳态，此时，如果我们在整个计算域的中间，即  $x = 10\text{m}$  的地方观察会发现，速度曲线几乎是平的。这条曲线可以通过在 paraFoam 中通过绘制  $U_x$  的曲线来得到。现在，用户可以将  $B_y$  的值调到 1，重新运行算例并运行  $U\text{components}$ 。可以看出， $M = 1$  的时候电磁力和粘性力大小相近。

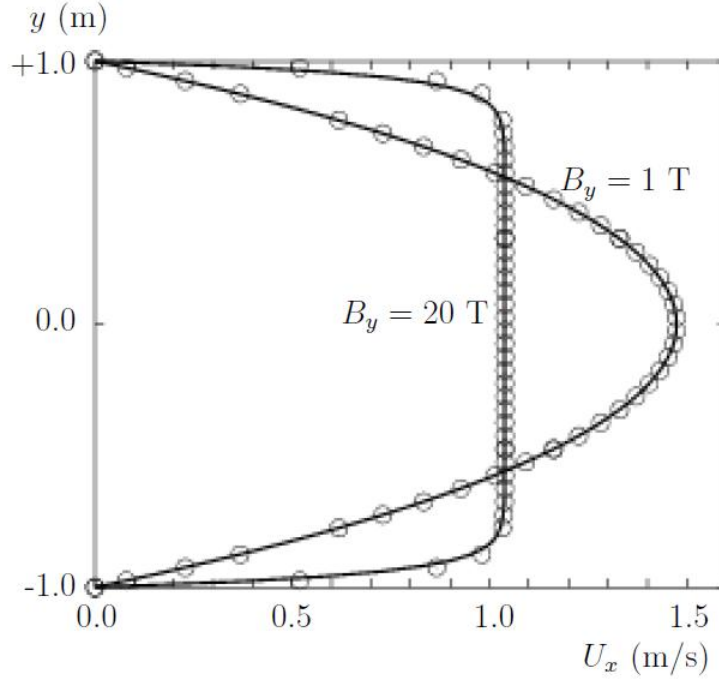


图 3.13 在  $B_y=1$  和  $B_y=10$  时 Hartmann 问题的速度曲线

这时，速度曲线看起来像泊肃叶流中的双曲线，如图 3.13 所示。为了验证数值解的正确性，解析解也被添加到图 3.13 中以作比较。解析解的表达式如下：

$$\frac{U_x(y)}{U_x(0)} = \frac{\cosh M - \cosh M(\frac{y}{L})}{\cosh M - 1} \quad (3.29)$$

这里，特征长度  $L$  为计算域宽度的一半，即  $1\text{m}$ 。



# 索引

操作符 数 A B C D E F G H I J K L M N O P Q R S T U V W X Z

## 操 作 符

\*  
    tensor member function, [P-21](#)  
+  
    tensor member function, [P-21](#)  
-  
    tensor member function, [P-21](#)  
/  
    tensor member function, [P-21](#)  
/\*...\*/  
    C++ syntax, [U-76](#)  
//  
    C++ syntax, [U-76](#)  
    OpenFOAM file syntax, [U-104](#)  
# include  
    C++ syntax, [U-70](#), [U-76](#)  
&  
    tensor member function, [P-21](#)  
&&  
    tensor member function, [P-21](#)  
^  
    tensor member function, [P-21](#)  
<LESModel>Coeffs keyword, [U-184](#)  
<RASModel>Coeffs keyword, [U-184](#)  
<delta>Coeffs keyword, [U-184](#)  
0.000000e+00 directory, [U-104](#)  
1-dimensional mesh, [U-130](#)  
1D mesh, [U-130](#)  
2-dimensional mesh, [U-130](#)  
2D mesh, [U-130](#)

## 数

0 directory, [U-104](#)

## A

access functions, [P-19](#)  
addLayersControls keyword, [U-146](#)  
adiabaticFlameT utility, [U-95](#)  
  
adiabaticPerfectFluid model, [U-99](#), [U-179](#)  
adjointShapeOptimizationFoam solver, [U-83](#)  
adjustableRunTime  
    keyword entry, [U-60](#), [U-111](#)  
adjustTimeStep keyword, [U-60](#), [U-112](#)  
agglomerator keyword, [U-123](#)  
algorithms tools, [U-96](#)  
alphaContactAngle  
    boundary condition, [U-57](#)  
analytical solution, [P-41](#)  
Animations window panel, [U-168](#)  
anisotropicFilter model, [U-100](#)  
Annotation window panel, [U-24](#), [U-167](#)  
ansysToFoam utility, [U-89](#)  
APIfunctions model, [U-99](#)  
applications, [U-67](#)  
Apply button, [U-164](#), [U-168](#)  
applyBoundaryLayer utility, [U-88](#)  
applyWallFunctionBoundaryConditions  
utility,  
    [U-88](#)  
arbitrarily unstructured, [P-27](#)  
arc  
    keyword entry, [U-139](#)  
arc keyword, [U-138](#)  
As keyword, [U-182](#)  
ascii  
    keyword entry, [U-112](#)  
attachMesh utility, [U-90](#)  
Auto Accept button, [U-168](#)  
autoMesh

- library, [U-96](#)
- autoPatch utility, [U-90](#)
- autoRefineMesh utility, [U-91](#)
- axes
  - right-handed, [U-136](#)
  - right-handed rectangular Cartesian, [P-11](#),  
[U-18](#)
- axi-symmetric cases, [U-135](#), [U-144](#)
- axi-symmetric mesh, [U-130](#)

## B

- background
  - process, [U-24](#), [U-79](#)
- backward
  - keyword entry, [U-119](#)
- Backward differencing, [P-35](#)
- barotropicCompressibilityModels
  - library, [U-98](#)
- basicMultiComponentMixture model, [U-98](#),  
[U-180](#)
- basicSolidThermo
  - library, [U-99](#)
- basicThermophysicalModels
  - library, [U-98](#)
- binary
  - keyword entry, [U-112](#)
- BirdCarreau model, [U-101](#)
- blended differencing, [P-34](#)
- block
  - expansion ratio, [U-140](#)
- block keyword, [U-138](#)
- blocking
  - keyword entry, [U-78](#)
- blockMesh
  - library, [U-96](#)
- blockMesh solver, [P-43](#)
- blockMesh utility, [U-37](#), [U-89](#), [U-136](#)
- blockMesh executable
  - vertex numbering, [U-140](#)
- blockMeshDict
  - dictionary, [U-18](#), [U-20](#), [U-35](#), [U-48](#), [U-136](#),  
[U-144](#)

- blocks keyword, [U-20](#), [U-30](#), [U-140](#)
- boundaries, [U-132](#)
- boundary, [U-132](#)
- boundary
  - dictionary, [U-129](#), [U-136](#)
- boundary keyword, [U-141](#)
- boundary condition
  - alphaContactAngle, [U-57](#)
  - buoyantPressure, [U-137](#)
  - calculated, [U-136](#)
  - cyclic, [U-135](#), [U-142](#)
  - directionMixed, [U-136](#)
  - empty, [P-59](#), [P-65](#), [U-18](#), [U-130](#), [U-135](#)
  - fixedGradient, [U-136](#)
  - fixedValue, [U-136](#)
  - fluxCorrectedVelocity, [U-137](#)
  - inlet, [P-65](#)
  - inletOutlet, [U-137](#)
  - mixed, [U-136](#)
  - movingWallVelocity, [U-137](#)
  - outlet, [P-65](#)
  - outletInlet, [U-137](#)
  - partialSlip, [U-137](#)
  - patch, [U-134](#)
  - pressureDirectedInletVelocity, [U-137](#)
  - pressureInletVelocity, [U-137](#)
  - pressureOutlet, [P-59](#)
  - pressureTransmissive, [U-137](#)
  - processor, [U-135](#)
  - setup, [U-20](#)
  - slip, [U-137](#)
  - supersonicFreeStream, [U-137](#)
  - surfaceNormalFixedValue, [U-137](#)
  - symmetryPlane, [P-59](#), [U-134](#)
  - totalPressure, [U-137](#)
  - turbulentInlet, [U-137](#)
  - wall, [U-40](#)
  - wall, [P-59](#), [P-65](#), [U-57](#), [U-134](#)
  - wedge, [U-130](#), [U-135](#), [U-144](#)
  - zeroGradient, [U-136](#)
- boundary conditions, [P-39](#)
  - Dirichlet, [P-39](#)
  - inlet, [P-40](#)
  - Neumann, [P-39](#)

- no-slip impermeable wall, [P-40](#)
- outlet, [P-40](#)
- physical, [P-40](#)
- symmetry plane, [P-40](#)
- boundaryField keyword, [U-21](#), [U-108](#)
- boundaryFoam solver, [U-84](#)
- bounded
  - keyword entry, [U-117](#), [U-118](#)
- boxToCell keyword, [U-58](#)
- boxTurb utility, [U-88](#)
- breaking of a dam, [U-55](#)
- buoyantBoussinesqPimpleFoam solver, [U-86](#)
- buoyantBoussinesqSimpleFoam solver, [U-86](#)
- buoyantPimpleFoam solver, [U-86](#)
- buoyantPressure
  - boundary condition, [U-137](#)
- buoyantSimpleFoam solver, [U-86](#)
- button
  - Apply, [U-164](#), [U-168](#)
  - Auto Accept, [U-168](#)
  - Choose Preset, [U-166](#)
  - Delete, [U-164](#)
  - Edit Color Map, [U-165](#)
  - Enable Line Series, [U-34](#)
  - Orientation Axes, [U-24](#), [U-167](#)
  - Refresh Times, [U-25](#)
  - Rescale to Data Range, [U-25](#)
  - Reset, [U-164](#)
  - Set Ambient Color, [U-166](#)
  - Update GUI, [U-165](#)
  - Use Parallel Projection, [U-24](#)
  - Use parallel projection, [U-167](#)

## C

- C++ syntax
  - `/*...*/`, [U-76](#)
  - `//`, [U-76](#)
  - `# include`, [U-70](#), [U-76](#)
- cacheAgglomeration keyword, [U-123](#)
- calculated
  - boundary condition, [U-136](#)
- cAlpha keyword, [U-61](#)

- cases, [U-103](#)
- castellatedMesh keyword, [U-146](#)
- castellatedMeshControls
  - dictionary, [U-147–U-149](#)
- castellatedMeshControls keyword, [U-146](#)
- cavitatingDyMFoam solver, [U-85](#)
- cavitatingFoam solver, [U-85](#)
- cavity flow, [U-17](#)
- ccm26ToFoam utility, [U-89](#)
- CEI ARCH
  - environment variable, [U-173](#)
- CEI HOME
  - environment variable, [U-173](#)
- cell
  - expansion ratio, [U-140](#)
- cell class, [P-27](#)
- cell
  - keyword entry, [U-174](#)
- cellLimited
  - keyword entry, [U-117](#)
- cellPoint
  - keyword entry, [U-174](#)
- cellPointFace
  - keyword entry, [U-174](#)
- cells
  - dictionary, [U-136](#)
- central differencing, [P-34](#)
- cfTools tools, [U-96](#)
- cfx4ToFoam utility, [U-89](#), [U-154](#)
- changeDictionary utility, [U-88](#)
- Charts window panel, [U-168](#)
- checkMesh utility, [U-90](#), [U-155](#)
- chemFoam solver, [U-86](#)
- chemistryModel
  - library, [U-99](#)
- chemistryModel model, [U-99](#)
- chemistrySolver model, [U-99](#)
- chemkinToFoam utility, [U-95](#)
- Choose Preset button, [U-166](#)
- chtMultiRegionSimpleFoam solver, [U-87](#)
- chtMultiRegionFoam solver, [U-87](#)
- Chung
  - library, [U-99](#)
- class

- cell, [P-27](#)
- dimensionSet, [P-21](#), [P-28](#), [P-29](#)
- face, [P-27](#)
- finiteVolumeCalculus, [P-29](#)
- finiteVolumeMethod, [P-29](#)
- fvMesh, [P-27](#)
- fvSchemes, [P-32](#)
- fvc, [P-32](#)
- fvm, [P-32](#)
- pointField, [P-27](#)
- polyBoundaryMesh, [P-27](#)
- polyMesh, [P-27](#), [U-127](#), [U-129](#)
- polyPatchList, [P-27](#)
- polyPatch, [P-27](#)
- scalarField, [P-25](#)
- scalar, [P-19](#)
- slice, [P-27](#)
- symmTensorField, [P-25](#)
- symmTensorThirdField, [P-25](#)
- tensorField, [P-25](#)
- tensorThirdField, [P-25](#)
- tensor, [P-19](#)
- vectorField, [P-25](#)
- vector, [P-19](#), [U-107](#)
- word, [P-21](#), [P-27](#)
- class keyword, [U-105](#)
- clockTime
  - keyword entry, [U-111](#)
- cloud keyword, [U-175](#)
- cloudFunctionObjects
  - library, [U-96](#)
- cmptAv
  - tensor member function, [P-21](#)
- Co utility, [U-92](#)
- coalChemistryFoam solver, [U-87](#)
- coalCombustion
  - library, [U-97](#)
- cofactors
  - tensor member function, [P-21](#)
- coldEngineFoam solver, [U-86](#)
- collapseEdges utility, [U-91](#)
- Color By menu, [U-166](#)
- Color Legend window, [U-27](#)
- Color Legend window panel, [U-166](#)
- Color Scale window panel, [U-166](#)
- Colors window panel, [U-168](#)
- compressibleInterDyMFoam solver, [U-85](#)
- compressibleInterFoam solver, [U-85](#)
- compressibleMultiphaseInterFoam solver, [U-85](#)
- combinePatchFaces utility, [U-91](#)
- comments, [U-76](#)
- commsType keyword, [U-78](#)
- compressed
  - keyword entry, [U-112](#)
- compressibleLESModels
  - library, [U-101](#)
- compressibleRASModels
  - library, [U-100](#)
- constant directory, [U-104](#), [U-179](#)
- constant model, [U-98](#)
- constTransport model, [U-99](#), [U-180](#)
- containers tools, [U-96](#)
- continuum
  - mechanics, [P-11](#)
- control
  - of time, [U-111](#)
- controlDict
  - dictionary, [P-61](#), [U-21](#), [U-30](#), [U-41](#), [U-50](#), [U-60](#), [U-104](#), [U-160](#)
- controlDict file, [P-46](#)
- convection, **see** divergence, [P-34](#)
- convergence, [U-38](#)
- conversion
  - library, [U-97](#)
- convertToMeters keyword, [U-138](#)
- coordinate
  - system, [P-11](#)
- coordinate system, [U-18](#)
- corrected
  - keyword entry, [U-117](#), [U-118](#)
- Courant number, [P-38](#), [U-22](#)
- Cp keyword, [U-181](#)
- cpuTime
  - keyword entry, [U-111](#)
- Crank Nicholson
  - temporal discretisation, [P-38](#)



CrankNicholson  
 keyword entry, [U-119](#)  
 createExternalCoupledPatchGeometry  
 utility,  
[U-88](#)  
 createBaffles utility, [U-90](#)  
 createPatch utility, [U-90](#)  
 createTurbulenceFields utility, [U-92](#)  
 cross product, **see** tensor, vector cross  
 product  
 CrossPowerLaw  
 keyword entry, [U-58](#)  
 CrossPowerLaw model, [U-101](#)  
 cubeRootVolDelta model, [U-101](#)  
 cubicCorrected  
 keyword entry, [U-119](#)  
 cubicCorrection  
 keyword entry, [U-116](#)  
 curl, [P-33](#)  
 curl  
 fvc member function, [P-33](#)  
 Current Time Controls menu, [U-25](#), [U-165](#)  
 curve keyword, [U-175](#)  
 Cv keyword, [U-181](#)  
 cyclic  
 boundary condition, [U-135](#), [U-142](#)  
 cyclic  
 keyword entry, [U-135](#)  
 cylinder  
 flow around a, [P-41](#)

## D

d2dt2  
 fvc member function, [P-33](#)  
 fvm member function, [P-33](#)  
 dam  
 breaking of a, [U-55](#)  
 datToFoam utility, [U-89](#)  
 db tools, [U-96](#)  
 ddt  
 fvc member function, [P-33](#)  
 fvm member function, [P-33](#)  
 DeardorffDiffStress model, [U-101](#)  
 debug keyword, [U-146](#)

decompose model, [U-97](#)  
 decomposePar utility, [U-79](#), [U-80](#), [U-95](#)  
 decomposeParDict  
 dictionary, [U-79](#)  
 decomposition  
 of field, [U-79](#)  
 of mesh, [U-79](#)  
 decompositionMethods  
 library, [U-97](#)  
 decompression of a tank, [P-58](#)  
 defaultFieldValues keyword, [U-58](#)  
 deformedGeom utility, [U-90](#)  
 Delete button, [U-164](#)  
 delta keyword, [U-81](#), [U-184](#)  
 deltaT keyword, [U-111](#)  
 dependencies, [U-70](#)  
 dependency lists, [U-70](#)  
 det  
 tensor member function, [P-21](#)  
 determinant, **see** tensor, determinant  
 dev  
 tensor member function, [P-21](#)  
 diag  
 tensor member function, [P-21](#)  
 diagonal  
 keyword entry, [U-121](#), [U-122](#)  
 DIC  
 keyword entry, [U-122](#)  
 DICGaussSeidel  
 keyword entry, [U-122](#)  
 dictionary  
 LESProperties, [U-184](#)  
 PISO, [U-23](#)  
 blockMeshDict, [U-18](#), [U-20](#), [U-35](#), [U-48](#),  
[U-136](#), [U-144](#)  
 boundary, [U-129](#), [U-136](#)  
 castellatedMeshControls, [U-147–U-149](#)  
 cells, [U-136](#)  
 controlDict, [P-61](#), [U-21](#), [U-30](#), [U-41](#),  
[U-50](#),  
[U-60](#), [U-104](#), [U-160](#)  
 decomposeParDict, [U-79](#)  
 faces, [U-129](#), [U-136](#)

- fvSchemes, [U-61](#), [U-104](#), [U-113](#)
- fvSolution, [U-104](#), [U-120](#)
- mechanicalProperties, [U-49](#)
- neighbour, [U-129](#)
- owner, [U-129](#)
- points, [U-129](#), [U-136](#)
- thermalProperties, [U-50](#)
- thermophysicalProperties, [U-179](#)
- transportProperties, [U-21](#), [U-38](#), [U-41](#)
- turbulenceProperties, [U-40](#), [U-59](#), [U-183](#)
- 183
  - differencing
    - Backward, [P-35](#)
    - blended, [P-34](#)
    - central, [P-34](#)
    - Euler implicit, [P-35](#)
    - Gamma, [P-34](#)
    - MINMOD, [P-34](#)
    - SUPERBEE, [P-34](#)
    - upwind, [P-34](#)
    - van Leer, [P-34](#)
  - DILU
    - keyword entry, [U-122](#)
  - dimension
    - checking in OpenFOAM, [P-21](#), [U-107](#)
  - dimensional units, [U-107](#)
  - dimensioned<Type> template class, [P-21](#)
  - dimensionedTypes tools, [U-96](#)
  - dimensions keyword, [U-20](#), [U-108](#)
  - dimensionSet class, [P-21](#), [P-28](#), [P-29](#)
  - dimensionSet tools, [U-96](#)
  - directionMixed
    - boundary condition, [U-136](#)
  - directory
    - 0.000000e+00, [U-104](#)
    - 0, [U-104](#)
    - Make, [U-71](#)
    - constant, [U-104](#), [U-179](#)
    - fluentInterface, [U-170](#)
    - polyMesh, [U-104](#), [U-129](#)
    - processorN, [U-80](#)
    - run, [U-103](#)
    - system, [P-46](#), [U-104](#)
    - tutorials, [P-41](#), [U-17](#)
  - discretisation
    - equation, [P-29](#)
  - Display window panel, [U-23](#), [U-25](#), [U-164](#), [U-165](#)
  - distance
    - keyword entry, [U-149](#), [U-175](#)
  - distributed model, [U-97](#)
  - distributed keyword, [U-81](#), [U-82](#)
  - distributionModels
    - library, [U-97](#)
  - div
    - fvc member function, [P-33](#)
    - fvm member function, [P-33](#)
    - divergence, [P-33](#), [P-35](#)
    - divSchemes keyword, [U-114](#)
    - dnsFoam solver, [U-86](#)
    - doLayers keyword, [U-146](#)
    - double inner product, *see* tensor,double
    - inner
      - product
    - DPMFoam solver, [U-87](#)
    - dsmc
      - library, [U-97](#)
    - dsmcFieldsCalc utility, [U-93](#)
    - dsmcFoam solver, [U-87](#)
    - dsmcInitialise utility, [U-88](#)
    - dx
      - keyword entry, [U-174](#)
    - dynamicFvMesh
      - library, [U-97](#)
    - dynamicMesh
      - library, [U-96](#)
    - dynLagrangian model, [U-101](#)
    - dynOneEqEddy model, [U-101](#)

## E

  - eConstThermo model, [U-99](#), [U-179](#)
  - edgeGrading keyword, [U-140](#)
  - edgeMesh
    - library, [U-97](#)
  - edges keyword, [U-138](#)
  - Edit menu, [U-167](#), [U-168](#)
  - Edit Color Map button, [U-165](#)
  - egrMixture model, [U-98](#), [U-180](#)

- electrostaticFoam solver, [U-88](#)
- empty
  - boundary condition, [P-59](#), [P-65](#), [U-18](#), [U-130](#), [U-135](#)
- empty
  - keyword entry, [U-135](#)
- Enable Line Series button, [U-34](#)
- endTime keyword, [U-22](#), [U-111](#)
- energy keyword, [U-181](#)
- engine
  - library, [U-97](#)
- engineCompRatio utility, [U-93](#)
- engineFoam solver, [U-86](#)
- engineSwirl utility, [U-88](#)
- ensight74FoamExec utility, [U-172](#)
- ENSIGHT7 INPUT
  - environment variable, [U-173](#)
- ENSIGHT7 READER
  - environment variable, [U-173](#)
- ensightFoamReader utility, [U-91](#)
- enstrophy utility, [U-92](#)
- environment variable
  - CEI ARCH, [U-173](#)
  - CEI HOME, [U-173](#)
  - ENSIGHT7 INPUT, [U-173](#)
  - ENSIGHT7 READER, [U-173](#)
  - FOAM RUN, [U-103](#)
  - WM ARCH OPTION, [U-74](#)
  - WM ARCH, [U-74](#)
  - WM COMPILER BIN, [U-74](#)
  - WM COMPILER DIR, [U-74](#)
  - WM COMPILER LIB, [U-74](#)
  - WM COMPILER, [U-74](#)
  - WM COMPILE OPTION, [U-74](#)
  - WM DIR, [U-74](#)
  - WM MPLIB, [U-74](#)
  - WM OPTIONS, [U-74](#)
  - WM PRECISION OPTION, [U-74](#)
  - WM PROJECT DIR, [U-74](#)
  - WM PROJECT INST DIR, [U-74](#)
  - WM PROJECT USER DIR, [U-74](#)
  - WM PROJECT VERSION, [U-74](#)
  - WM PROJECT, [U-74](#)
  - wmake, [U-73](#)

- equationOfState keyword, [U-181](#)
- equilibriumCO utility, [U-95](#)
- equilibriumFlameT utility, [U-95](#)
- errorReduction keyword, [U-153](#)
- Euler
  - keyword entry, [U-119](#)
- Euler implicit
  - differencing, [P-35](#)
  - temporal discretisation, [P-38](#)
- examples
  - decompression of a tank, [P-58](#)
  - flow around a cylinder, [P-41](#)
  - flow over backward step, [P-49](#)
  - Hartmann problem, [P-63](#)
  - supersonic flow over forward step, [P-54](#)
- execFlowFunctionObjects utility, [U-93](#)
- expandDictionary utility, [U-95](#)
- expansionRatio keyword, [U-152](#)
- explicit
  - temporal discretisation, [P-38](#)
- extrude2DMesh utility, [U-89](#)
- extrudeMesh utility, [U-89](#)
- extrudeToRegionMesh utility, [U-89](#)

## F

- face class, [P-27](#)
- face keyword, [U-175](#)
- faceAgglomerate utility, [U-88](#)
- faceAreaPair
  - keyword entry, [U-123](#)
- faceLimited
  - keyword entry, [U-117](#)
- faces
  - dictionary, [U-129](#), [U-136](#)
- FDIC
  - keyword entry, [U-122](#)
- featureAngle keyword, [U-152](#)
- features keyword, [U-147](#), [U-148](#)
- field
  - U, [U-22](#)
  - p, [U-22](#)
  - decomposition, [U-79](#)
- FieldField<Type> template class, [P-28](#)

- fieldFunctionObjects
  - library, [U-96](#)
- fields, [P-25](#)
  - mapping, [U-160](#)
- fields tools, [U-96](#)
- fields keyword, [U-174](#)
- Field<Type> template class, [P-25](#)
- fieldValues keyword, [U-58](#)
- file
  - Make/files, [U-72](#)
  - controlDict, [P-46](#)
  - files, [U-71](#)
  - g, [U-59](#)
  - options, [U-71](#)
  - snappyHexMeshDict, [U-145](#)
  - transportProperties, [U-58](#)
- file format, [U-104](#)
- fileFormats
  - library, [U-97](#)
- fileModificationChecking keyword, [U-78](#)
- fileModificationSkew keyword, [U-78](#)
- files file, [U-71](#)
- filteredLinear2
  - keyword entry, [U-116](#)
- finalLayerThickness keyword, [U-152](#)
- financialFoam solver, [U-88](#)
- finite volume
  - discretisation, [P-23](#)
  - mesh, [P-27](#)
- finiteVolume
  - library, [U-96](#)
- finiteVolume tools, [U-96](#)
- finiteVolumeCalculus class, [P-29](#)
- finiteVolumeMethod class, [P-29](#)
- fireFoam solver, [U-86](#)
- firstTime keyword, [U-111](#)
- fixed
  - keyword entry, [U-112](#)
- fixedGradient
  - boundary condition, [U-136](#)
- fixedValue
  - boundary condition, [U-136](#)
- flattenMesh utility, [U-90](#)
- floatTransfer keyword, [U-78](#)
- flow
  - free surface, [U-55](#)
  - laminar, [U-17](#)
  - steady, turbulent, [P-49](#)
  - supersonic, [P-55](#)
  - turbulent, [U-17](#)
- flow around a cylinder, [P-41](#)
- flow over backward step, [P-49](#)
- flowType utility, [U-92](#)
- fluent3DMeshToFoam utility, [U-89](#)
- fluentInterface directory, [U-170](#)
- fluentMeshToFoam utility, [U-89](#), [U-154](#)
- fluxCorrectedVelocity
  - boundary condition, [U-137](#)
- fluxRequired keyword, [U-114](#)
- OpenFOAM
  - cases, [U-103](#)
- FOAM RUN
  - environment variable, [U-103](#)
- foamCalc utility, [U-33](#), [U-93](#)
- foamCalcFunctions
  - library, [U-96](#)
- foamCorrectVrt script/alias, [U-158](#)
- foamDataToFluent utility, [U-91](#), [U-170](#)
- foamDebugSwitches utility, [U-95](#)
- FoamFile keyword, [U-105](#)
- foamFile
  - keyword entry, [U-174](#)
- foamFormatConvert utility, [U-95](#)
- foamHelp utility, [U-95](#)
- foamInfoExec utility, [U-95](#)
- foamJob script/alias, [U-177](#)
- foamListTimes utility, [U-93](#)
- foamLog script/alias, [U-177](#)
- foamMeshToFluent utility, [U-89](#), [U-170](#)
- foamToEnight utility, [U-91](#)
- foamToEnightParts utility, [U-91](#)
- foamToGMV utility, [U-91](#)
- foamToStarMesh utility, [U-89](#)
- foamToSurface utility, [U-89](#)
- foamToTecplot360 utility, [U-91](#)
- foamToVTK utility, [U-91](#)
- foamUpgradeCyclics utility, [U-88](#)
- foamUpgradeFvSolution utility, [U-88](#)

- foamyHexMeshBackgroundMesh utility, [U-89](#)
- foamyHexMeshSurfaceSimplify utility, [U-89](#)
- foamyHexMesh utility, [U-89](#)
- foamyQuadMesh utility, [U-89](#)
- forces
  - library, [U-96](#)
- foreground
  - process, [U-24](#)
- format keyword, [U-105](#)
- fourth
  - keyword entry, [U-117](#), [U-118](#)
- functions keyword, [U-113](#)
- fvc class, [P-32](#)
- fvc member function
  - curl, [P-33](#)
  - d2dt2, [P-33](#)
  - ddt, [P-33](#)
  - div, [P-33](#)
  - gGrad, [P-33](#)
  - grad, [P-33](#)
  - laplacian, [P-33](#)
  - lsGrad, [P-33](#)
  - snGrad, [P-33](#)
  - snGradCorrection, [P-33](#)
  - sqrGradGrad, [P-33](#)
- fvDOM
  - library, [U-98](#)
- FVFunctionObjects
  - library, [U-96](#)
- fvm class, [P-32](#)
- fvm member function
  - d2dt2, [P-33](#)
  - ddt, [P-33](#)
  - div, [P-33](#)
  - laplacian, [P-33](#)
  - Su, [P-33](#)
  - SuSp, [P-33](#)
- fvMatrices tools, [U-96](#)
- fvMatrix template class, [P-29](#)
- fvMesh class, [P-27](#)
- fvMesh tools, [U-96](#)
- fvMotionSolvers

- library, [U-97](#)
- fvSchemes
  - dictionary, [U-61](#), [U-104](#), [U-113](#)
- fvSchemes class, [P-32](#)
- fvSchemes
  - menu entry, [U-51](#)
- fvSolution
  - dictionary, [U-104](#), [U-120](#)

## G

- g file, [U-59](#)
- gambitToFoam utility, [U-89](#), [U-154](#)
- GAMG
  - keyword entry, [U-52](#), [U-121](#), [U-122](#)
- Gamma
  - keyword entry, [U-116](#)
- Gamma differencing, [P-34](#)
- Gauss
  - keyword entry, [U-117](#)
- Gauss's theorem, [P-32](#)
- GaussSeidel
  - keyword entry, [U-122](#)
- General window panel, [U-167](#), [U-168](#)
- general
  - keyword entry, [U-112](#)
- genericFvPatchField
  - library, [U-97](#)
- geometric-algebraic multi-grid, [U-122](#)
- GeometricBoundaryField template class, [P-28](#)
- geometricField<Type> template class, [P-28](#)
- geometry keyword, [U-146](#)
- gGrad
  - fvc member function, [P-33](#)
- global tools, [U-96](#)
- gmshToFoam utility, [U-89](#)
- gnuplot
  - keyword entry, [U-112](#), [U-174](#)
- grad
  - fvc member function, [P-33](#)
- (Grad Grad) squared, [P-33](#)
- gradient, [P-33](#), [P-36](#)
  - Gauss scheme, [P-36](#)
  - Gauss's theorem, [U-51](#)

- least square fit, [U-51](#)
- least squares method, [P-36](#), [U-51](#)
- surface normal, [P-36](#)
- gradSchemes keyword, [U-114](#)
- graph tools, [U-96](#)
- graphFormat keyword, [U-112](#)
- GuldersEGRLaminarFlameSpeed model, [U-98](#)
- GuldersLaminarFlameSpeed model, [U-98](#)

## H

- hConstThermo model, [U-99](#), [U-179](#)
- heheupsiReactionThermo model, [U-98](#), [U-180](#)
- Help menu, [U-167](#)
- hePsiThermo model, [U-98](#), [U-180](#)
- heRhoThermo model, [U-98](#), [U-180](#)
- HerschelBulkley model, [U-101](#)
- hExponentialThermo
  - library, [U-99](#)
- Hf keyword, [U-181](#)
- hierarchical
  - keyword entry, [U-80](#), [U-81](#)
- highCpCoeffs keyword, [U-182](#)
- homogenousDynOneEqEddy model, [U-101](#)
- homogenousDynSmagorinsky model, [U-101](#)
- homogeneousMixture model, [U-98](#), [U-180](#)
- hPolynomialThermo model, [U-99](#), [U-180](#)

## I

- I
  - tensor member function, [P-21](#)
- icoFoam solver, [U-17](#), [U-21](#), [U-22](#), [U-24](#), [U-84](#)
- icoPolynomial model, [U-99](#), [U-179](#)
- icoUncoupledKinematicParcelDyMFoam solver,
  - [U-87](#)
- icoUncoupledKinematicParcelFoam solver, [U-87](#)
- ideasToFoam utility, [U-154](#)
- ideasUnvToFoam utility, [U-89](#)
- identities, [see](#) tensor, identities

- identity, [see](#) tensor, identity
- incompressibleLESModels
  - library, [U-101](#)
- incompressiblePerfectGas model, [U-99](#), [U-179](#)
- incompressibleRASModels
  - library, [U-100](#)
- incompressibleTransportModels
  - library, [P-50](#), [U-101](#)
- incompressibleTurbulenceModels
  - library, [P-50](#)
- index
  - notation, [P-12](#), [P-13](#)
- Information window panel, [U-164](#)
- inhomogeneousMixture model, [U-98](#), [U-180](#)
- inlet
  - boundary condition, [P-65](#)
- inletOutlet
  - boundary condition, [U-137](#)
- inner product, [see](#) tensor, inner product
- inotify
  - keyword entry, [U-78](#)
- inotifyMaster
  - keyword entry, [U-78](#)
- inside
  - keyword entry, [U-149](#)
- insideCells utility, [U-90](#)
- interPhaseChangeDyMFoam solver, [U-85](#)
- interPhaseChangeFoam solver, [U-85](#)
- interDyMFoam solver, [U-85](#)
- interfaceProperties
  - library, [U-102](#)
- interfaceProperties model, [U-102](#)
- interFoam solver, [U-85](#)
- interMixingFoam solver, [U-85](#)
- internalField keyword, [U-21](#), [U-108](#)
- interpolation tools, [U-96](#)
- interpolationScheme keyword, [U-174](#)
- interpolations tools, [U-96](#)
- interpolationSchemes keyword, [U-114](#)
- inv
  - tensor member function, [P-21](#)
- iterations

maximum, [U-121](#)

## J

janafThermo model, [U-99](#), [U-180](#)

jobControl

library, [U-96](#)

jplot

keyword entry, [U-112](#), [U-174](#)

## K

kEpsilon model, [U-100](#)

keyword

As, [U-182](#)

Cp, [U-181](#)

Cv, [U-181](#)

FoamFile, [U-105](#)

Hf, [U-181](#)

LESModel, [U-184](#)

Pr, [U-182](#)

RASModel, [U-184](#)

Tcommon, [U-182](#)

Thigh, [U-182](#)

Tlow, [U-182](#)

Ts, [U-182](#)

addLayersControls, [U-146](#)

adjustTimeStep, [U-60](#), [U-112](#)

agglomerator, [U-123](#)

arc, [U-138](#)

blocks, [U-20](#), [U-30](#), [U-140](#)

block, [U-138](#)

boundaryField, [U-21](#), [U-108](#)

boundary, [U-141](#)

boxToCell, [U-58](#)

cAlpha, [U-61](#)

cacheAgglomeration, [U-123](#)

castellatedMeshControls, [U-146](#)

castellatedMesh, [U-146](#)

class, [U-105](#)

cloud, [U-175](#)

commsType, [U-78](#)

convertToMeters, [U-138](#)

curve, [U-175](#)

debug, [U-146](#)

defaultFieldValues, [U-58](#)

deltaT, [U-111](#)

delta, [U-81](#), [U-184](#)

dimensions, [U-20](#), [U-108](#)

distributed, [U-81](#), [U-82](#)

divSchemes, [U-114](#)

doLayers, [U-146](#)

edgeGrading, [U-140](#)

edges, [U-138](#)

endTime, [U-22](#), [U-111](#)

energy, [U-181](#)

equationOfState, [U-181](#)

errorReduction, [U-153](#)

expansionRatio, [U-152](#)

face, [U-175](#)

featureAngle, [U-152](#)

features, [U-147](#), [U-148](#)

fieldValues, [U-58](#)

fields, [U-174](#)

fileModificationChecking, [U-78](#)

fileModificationSkew, [U-78](#)

finalLayerThickness, [U-152](#)

firstTime, [U-111](#)

floatTransfer, [U-78](#)

fluxRequired, [U-114](#)

format, [U-105](#)

functions, [U-113](#)

geometry, [U-146](#)

gradSchemes, [U-114](#)

graphFormat, [U-112](#)

highCpCoeffs, [U-182](#)

internalField, [U-21](#), [U-108](#)

interpolationSchemes, [U-114](#)

interpolationScheme, [U-174](#)

laplacianSchemes, [U-114](#)

latestTime, [U-38](#)

layers, [U-152](#)

leastSquares, [U-51](#)

levels, [U-150](#)

libs, [U-78](#), [U-112](#)

locationInMesh, [U-148](#), [U-149](#)

location, [U-105](#)

lowCpCoeffs, [U-182](#)

manualCoeffs, [U-81](#)

maxAlphaCo, [U-60](#)

maxBoundarySkewness, [U-153](#)  
 maxConcave, [U-153](#)  
 maxCo, [U-60](#), [U-112](#)  
 maxDeltaT, [U-60](#)  
 maxFaceThicknessRatio, [U-152](#)  
 maxGlobalCells, [U-148](#)  
 maxInternalSkewness, [U-153](#)  
 maxIter, [U-121](#)  
 maxLocalCells, [U-148](#)  
 maxNonOrtho, [U-153](#)  
 maxThicknessToMedialRatio, [U-152](#)  
 mergeLevels, [U-123](#)  
 mergePatchPairs, [U-138](#)  
 mergeTolerance, [U-146](#)  
 meshQualityControls, [U-146](#)  
 method, [U-81](#)  
 midPointAndFace, [U-175](#)  
 midPoint, [U-175](#)  
 minArea, [U-153](#)  
 minDeterminant, [U-153](#)  
 minFaceWeight, [U-153](#)  
 minFlatness, [U-153](#)  
 minMedianAxisAngle, [U-152](#)  
 minRefinementCells, [U-148](#)  
 minThickness, [U-152](#)  
 minTriangleTwist, [U-153](#)  
 minTwist, [U-153](#)  
 minVolRatio, [U-153](#)  
 minVol, [U-153](#)  
 mode, [U-149](#)  
 molWeight, [U-181](#)  
 mu, [U-182](#)  
 nAlphaSubCycles, [U-61](#)  
 nBufferCellsNoExtrude, [U-152](#)  
 nCellsBetweenLevels, [U-148](#)  
 nFaces, [U-130](#)  
 nFinestSweeps, [U-123](#)  
 nGrow, [U-152](#)  
 nLayerIter, [U-152](#)  
 nMoles, [U-181](#)  
 nPostSweeps, [U-123](#)  
 nPreSweeps, [U-123](#)  
 nRelaxIter, [U-150](#), [U-152](#)  
 nRelaxedIter, [U-152](#)  
 nSmoothNormals, [U-152](#)  
 nSmoothPatch, [U-150](#)  
 nSmoothScale, [U-153](#)  
 nSmoothSurfaceNormals, [U-152](#)  
 nSmoothThickness, [U-152](#)  
 nSolveIter, [U-150](#)  
 neighbourPatch, [U-142](#)  
 numberOfSubdomains, [U-81](#)  
 n, [U-81](#)  
 object, [U-105](#)  
 order, [U-81](#)  
 pRefCell, [U-23](#), [U-124](#)  
 pRefValue, [U-23](#), [U-124](#)  
 p rhgRefCell, [U-124](#)  
 p rhgRefValue, [U-124](#)  
 patchMap, [U-160](#)  
 patches, [U-138](#)  
 preconditioner, [U-121](#), [U-122](#)  
 pressure, [U-49](#)  
 printCoeffs, [U-41](#), [U-184](#)  
 processorWeights, [U-80](#)  
 processorWeights, [U-81](#)  
 purgeWrite, [U-112](#)  
 refGradient, [U-136](#)  
 refinementRegions, [U-148](#), [U-150](#)  
 refinementSurfaces, [U-148](#)  
 refinementRegions, [U-149](#)  
 regions, [U-58](#)  
 relTol, [U-52](#), [U-121](#)  
 relativeSizes, [U-152](#)  
 relaxed, [U-153](#)  
 resolveFeatureAngle, [U-148](#)  
 roots, [U-81](#), [U-82](#)  
 runTimeModifiable, [U-112](#)  
 scotchCoeffs, [U-81](#)  
 setFormat, [U-174](#)  
 sets, [U-174](#)  
 simpleGrading, [U-140](#)  
 simulationType, [U-40](#), [U-59](#), [U-183](#)  
 smoother, [U-123](#)  
 snGradSchemes, [U-114](#)  
 snapControls, [U-146](#)  
 snap, [U-146](#)  
 solvers, [U-120](#)



solver, [U-52](#), [U-121](#)  
 specie, [U-181](#)  
 spline, [U-138](#)  
 startFace, [U-130](#)  
 startFrom, [U-22](#), [U-111](#)  
 startTime, [U-22](#), [U-111](#)  
 stopAt, [U-111](#)  
 strategy, [U-80](#), [U-81](#)  
 surfaceFormat, [U-174](#)  
 surfaces, [U-174](#)  
 thermoType, [U-179](#)  
 thermodynamics, [U-181](#)  
 timeFormat, [U-112](#)  
 timePrecision, [U-112](#)  
 timeScheme, [U-114](#)  
 tolerance, [U-52](#), [U-121](#), [U-150](#)  
 topoSetSource, [U-58](#)  
 traction, [U-49](#)  
 transport, [U-181](#)  
 turbulence, [U-184](#)  
 type, [U-132](#), [U-133](#)  
 uniform, [U-175](#)  
 valueFraction, [U-136](#)  
 value, [U-21](#), [U-136](#)  
 version, [U-105](#)  
 vertices, [U-20](#), [U-138](#), [U-139](#)  
 writeCompression, [U-112](#)  
 writeControl, [U-22](#), [U-60](#), [U-111](#)  
 writeFormat, [U-54](#), [U-112](#)  
 writeInterval, [U-22](#), [U-31](#), [U-111](#)  
 writePrecision, [U-112](#)  
 <LESModel>Coeffs, [U-184](#)  
 <RASModel>Coeffs, [U-184](#)  
 <delta>Coeffs, [U-184](#)  
 keyword entry  
     CrankNicholson, [U-119](#)  
     CrossPowerLaw, [U-58](#)  
     DICGaussSeidel, [U-122](#)  
     DIC, [U-122](#)  
     DILU, [U-122](#)  
     Euler, [U-119](#)  
     FDIC, [U-122](#)  
     GAMG, [U-52](#), [U-121](#), [U-122](#)  
     Gamma, [U-116](#)  
     GaussSeidel, [U-122](#)  
     Gauss, [U-117](#)  
     LESModel, [U-40](#), [U-183](#)  
     MGridGen, [U-123](#)  
     MUSCL, [U-116](#)  
     Newtonian, [U-58](#)  
     PBiCG, [U-121](#)  
     PCG, [U-121](#)  
     QUICK, [U-119](#)  
     RASModel, [U-40](#), [U-183](#)  
     SFCD, [U-116](#), [U-119](#)  
     UMIST, [U-115](#)  
     adjustableRunTime, [U-60](#), [U-111](#)  
     arc, [U-139](#)  
     ascii, [U-112](#)  
     backward, [U-119](#)  
     binary, [U-112](#)  
     blocking, [U-78](#)  
     bounded, [U-117](#), [U-118](#)  
     cellLimited, [U-117](#)  
     cellPointFace, [U-174](#)  
     cellPoint, [U-174](#)  
     cell, [U-174](#)  
     clockTime, [U-111](#)  
     compressed, [U-112](#)  
     corrected, [U-117](#), [U-118](#)  
     cpuTime, [U-111](#)  
     cubicCorrected, [U-119](#)  
     cubicCorrection, [U-116](#)  
     cyclic, [U-135](#)  
     diagonal, [U-121](#), [U-122](#)  
     distance, [U-149](#), [U-175](#)  
     dx, [U-174](#)  
     empty, [U-135](#)  
     faceAreaPair, [U-123](#)  
     faceLimited, [U-117](#)  
     filteredLinear2, [U-116](#)  
     fixed, [U-112](#)  
     foamFile, [U-174](#)  
     fourth, [U-117](#), [U-118](#)  
     general, [U-112](#)  
     gnuplot, [U-112](#), [U-174](#)  
     hierarchical, [U-80](#), [U-81](#)  
     inotifyMaster, [U-78](#)

- inotify, [U-78](#)
- inside, [U-149](#)
- jplot, [U-112](#), [U-174](#)
- laminar, [U-40](#), [U-183](#)
- latestTime, [U-111](#)
- leastSquares, [U-117](#)
- limitedCubic, [U-116](#)
- limitedLinear, [U-116](#)
- limited, [U-117](#), [U-118](#)
- linearUpwind, [U-116](#), [U-119](#)
- linear, [U-116](#), [U-119](#)
- line, [U-139](#)
- localEuler, [U-119](#)
- manual, [U-80](#), [U-81](#)
- metis, [U-81](#)
- midPoint, [U-116](#)
- nextWrite, [U-111](#)
- noWriteNow, [U-111](#)
- nonBlocking, [U-78](#)
- none, [U-115](#), [U-122](#)
- null, [U-174](#)
- outside, [U-149](#)
- patch, [U-135](#), [U-176](#)
- polyLine, [U-139](#)
- polySpline, [U-139](#)
- processor, [U-135](#)
- raw, [U-112](#), [U-174](#)
- runTime, [U-31](#), [U-111](#)
- scheduled, [U-78](#)
- scientific, [U-112](#)
- scotch, [U-80](#), [U-81](#)
- simpleSpline, [U-139](#)
- simple, [U-80](#), [U-81](#)
- skewLinear, [U-116](#), [U-119](#)
- smoothSolver, [U-121](#)
- startTime, [U-22](#), [U-111](#)
- steadyState, [U-119](#)
- stl, [U-174](#)
- symmetryPlane, [U-135](#)
- timeStampMaster, [U-78](#)
- timeStamp, [U-78](#)
- timeStep, [U-22](#), [U-31](#), [U-111](#)
- uncompressed, [U-112](#)
- uncorrected, [U-117](#), [U-118](#)

- upwind, [U-116](#), [U-119](#)
- vanLeer, [U-116](#)
- vtk, [U-174](#)
- wall, [U-135](#)
- wedge, [U-135](#)
- writeControl, [U-111](#)
- writeNow, [U-111](#)
- xmgr, [U-112](#), [U-174](#)
- xyz, [U-175](#)
- x, [U-175](#)
- y, [U-175](#)
- z, [U-175](#)

- kivaToFoam utility, [U-89](#)
- kkLOmega model, [U-100](#)
- kOmega model, [U-100](#)
- kOmegaSST model, [U-100](#)
- kOmegaSSTSAS model, [U-101](#)
- Kronecker delta, [P-16](#)

## L

- lagrangian
  - library, [U-97](#)
- lagrangianIntermediate
  - library, [U-97](#)
- Lambda2 utility, [U-92](#)
- LamBremhorstKE model, [U-100](#)
- laminar model, [U-100](#), [U-101](#)
- laminar
  - keyword entry, [U-40](#), [U-183](#)
- laminarFlameSpeedModels
  - library, [U-98](#)
- laplaceFilter model, [U-100](#)
- Laplacian, [P-34](#)
- laplacian, [P-33](#)
- laplacian
  - fvc member function, [P-33](#)
  - fvm member function, [P-33](#)
- laplacianFoam solver, [U-83](#)
- laplacianSchemes keyword, [U-114](#)
- latestTime
  - keyword entry, [U-111](#)
- latestTime keyword, [U-38](#)
- LaundryGibsonRSTM model, [U-100](#)
- LaundrySharmaKE model, [U-100](#)

- layers keyword, [U-152](#)
- leastSquares
  - keyword entry, [U-117](#)
- leastSquares keyword, [U-51](#)
- LESdeltas
  - library, [U-100](#)
- LESfilters
  - library, [U-100](#)
- LESModel
  - keyword entry, [U-40](#), [U-183](#)
- LESModel keyword, [U-184](#)
- LESProperties
  - dictionary, [U-184](#)
- levels keyword, [U-150](#)
- libraries, [U-67](#)
- library
  - Chung, [U-99](#)
  - FVFunctionObjects, [U-96](#)
  - LESdeltas, [U-100](#)
  - LESfilters, [U-100](#)
  - MGridGenGAMGAgglomeration, [U-97](#)
  - ODE, [U-97](#)
  - OSspecific, [U-97](#)
  - OpenFOAM, [U-96](#)
  - P1, [U-98](#)
  - PV3FoamReader, [U-163](#)
  - PVFoamReader, [U-163](#)
  - SLGThermo, [U-99](#)
  - Wallis, [U-99](#)
  - autoMesh, [U-96](#)
  - barotropicCompressibilityModels, [U-98](#)
  - basicSolidThermo, [U-99](#)
  - basicThermophysicalModels, [U-98](#)
  - blockMesh, [U-96](#)
  - chemistryModel, [U-99](#)
  - cloudFunctionObjects, [U-96](#)
  - coalCombustion, [U-97](#)
  - compressibleLESModels, [U-101](#)
  - compressibleRASModels, [U-100](#)
  - conversion, [U-97](#)
  - decompositionMethods, [U-97](#)
  - distributionModels, [U-97](#)
  - dsmc, [U-97](#)
  - dynamicFvMesh, [U-97](#)
  - dynamicMesh, [U-96](#)
  - edgeMesh, [U-97](#)
  - engine, [U-97](#)
  - fieldFunctionObjects, [U-96](#)
  - fileFormats, [U-97](#)
  - finiteVolume, [U-96](#)
  - foamCalcFunctions, [U-96](#)
  - forces, [U-96](#)
  - fvDOM, [U-98](#)
  - fvMotionSolvers, [U-97](#)
    - genericFvPatchField, [U-97](#)
  - hExponentialThermo, [U-99](#)
  - incompressibleLESModels, [U-101](#)
  - incompressibleRASModels, [U-100](#)
  - incompressibleTransportModels, [P-50](#), [U-101](#)
  - incompressibleTurbulenceModels, [P-50](#)
  - interfaceProperties, [U-102](#)
  - jobControl, [U-96](#)
  - lagrangianIntermediate, [U-97](#)
  - lagrangian, [U-97](#)
  - laminarFlameSpeedModels, [U-98](#)
  - linear, [U-99](#)
  - liquidMixtureProperties, [U-99](#)
  - liquidProperties, [U-99](#)
  - meshTools, [U-97](#)
  - molecularMeasurements, [U-97](#)
  - molecule, [U-97](#)
  - opaqueSolid, [U-98](#)
  - pairPatchAgglomeration, [U-97](#)
  - postCalc, [U-96](#)
  - potential, [U-97](#)
  - primitive, [P-19](#)
  - radiationModels, [U-98](#)
  - randomProcesses, [U-97](#)
  - reactionThermophysicalModels, [U-98](#)
  - sampling, [U-96](#)
  - solidChemistryModel, [U-99](#)
  - solidMixtureProperties, [U-100](#)
  - solidParticle, [U-97](#)
  - solidProperties, [U-99](#)
  - solidSpecie, [U-100](#)

- solidThermo, [U-100](#)
- specie, [U-99](#)
- spray, [U-97](#)
- surfMesh, [U-97](#)
- surfaceFilmModels, [U-102](#)
- systemCall, [U-96](#)
- thermophysicalFunctions, [U-99](#)
- thermophysical, [U-179](#)
- topoChangerFvMesh, [U-97](#)
- triSurface, [U-97](#)
- turbulence, [U-97](#)
- twoPhaseProperties, [U-102](#)
- utilityFunctionObjects, [U-96](#)
- viewFactor, [U-98](#)
- vtkFoam, [U-163](#)
- vtkPV3Foam, [U-163](#)
- libs keyword, [U-78](#), [U-112](#)
- lid-driven cavity flow, [U-17](#)
- LienCubicKE model, [U-100](#)
- LienCubicKELowRe model, [U-100](#)
- LienLeschzinerLowRe model, [U-100](#)
- Lights window panel, [U-167](#)
- limited
  - keyword entry, [U-117](#), [U-118](#)
- limitedCubic
  - keyword entry, [U-116](#)
- limitedLinear
  - keyword entry, [U-116](#)
- line
  - keyword entry, [U-139](#)
- Line Style menu, [U-34](#)
- linear
  - library, [U-99](#)
- linear
  - keyword entry, [U-116](#), [U-119](#)
- linearUpwind
  - keyword entry, [U-116](#), [U-119](#)
- liquid
  - electrically-conducting, [P-63](#)
- liquidMixtureProperties
  - library, [U-99](#)
- liquidProperties
  - library, [U-99](#)
- lists, [P-25](#)

- List<Type> template class, [P-25](#)
- localEuler
  - keyword entry, [U-119](#)
- location keyword, [U-105](#)
- locationInMesh keyword, [U-148](#), [U-149](#)
- lowCpCoeffs keyword, [U-182](#)
- lowReOneEqEddy model, [U-101](#)
- LRDDiffStress model, [U-101](#)
- LRR model, [U-100](#)
- lsGrad
  - fvc member function, [P-33](#)
- LTSInterFoam solver, [U-85](#)
- LTSReactingFoam solver, [U-86](#)
- LTSReactingParcelFoam solver, [U-87](#)

## M

- Mach utility, [U-92](#)
- mag
  - tensor member function, [P-21](#)
- magneticFoam solver, [U-88](#)
- magnetohydrodynamics, [P-63](#)
- magSqr
  - tensor member function, [P-21](#)
- Make directory, [U-71](#)
- make script/alias, [U-69](#)
- Make/files file, [U-72](#)
- manual
  - keyword entry, [U-80](#), [U-81](#)
- manualCoeffs keyword, [U-81](#)
- mapFields utility, [U-30](#), [U-37](#), [U-41](#), [U-54](#), [U-89](#), [U-160](#)
- mapping
  - fields, [U-160](#)
- Marker Style menu, [U-34](#)
- matrices tools, [U-96](#)
- max
  - tensor member function, [P-21](#)
- maxAlphaCo keyword, [U-60](#)
- maxBoundarySkewness keyword, [U-153](#)
- maxCo keyword, [U-60](#), [U-112](#)
- maxConcave keyword, [U-153](#)
- maxDeltaT keyword, [U-60](#)
- maxDeltaxyz model, [U-101](#)

- maxFaceThicknessRatio keyword, [U-152](#)
- maxGlobalCells keyword, [U-148](#)
- maximum iterations, [U-121](#)
- maxInternalSkewness keyword, [U-153](#)
- maxIter keyword, [U-121](#)
- maxLocalCells keyword, [U-148](#)
- maxNonOrtho keyword, [U-153](#)
- maxThicknessToMedialRatio keyword, [U-152](#)
- mdEquilibrationFoam solver, [U-87](#)
- mdFoam solver, [U-87](#)
- mdInitialise utility, [U-89](#)
- mechanicalProperties
  - dictionary, [U-49](#)
- memory tools, [U-96](#)
- menu
  - Color By, [U-166](#)
  - Current Time Controls, [U-25](#), [U-165](#)
  - Edit, [U-167](#), [U-168](#)
  - Help, [U-167](#)
  - Line Style, [U-34](#)
  - Marker Style, [U-34](#)
  - VCR Controls, [U-25](#), [U-165](#)
  - View, [U-167](#)
- menu entry
  - Plot Over Line, [U-34](#)
  - Save Animation, [U-169](#)
  - Save Screenshot, [U-169](#)
  - Settings, [U-168](#)
  - Show Color Legend, [U-25](#)
  - Solid Color, [U-166](#)
  - Toolbars, [U-167](#)
  - View Settings..., [U-24](#)
  - View Settings, [U-24](#), [U-167](#)
  - Wireframe, [U-166](#)
  - fvSchemes, [U-51](#)
- mergeLevels keyword, [U-123](#)
- mergeMeshes utility, [U-90](#)
- mergeOrSplitBaffles utility, [U-90](#)
- mergePatchPairs keyword, [U-138](#)
- mergeTolerance keyword, [U-146](#)
- mesh
  - 1-dimensional, [U-130](#)
  - 1D, [U-130](#)
  - 2-dimensional, [U-130](#)
  - 2D, [U-130](#)
  - axi-symmetric, [U-130](#)
  - basic, [P-27](#)
  - block structured, [U-136](#)
  - decomposition, [U-79](#)
  - description, [U-127](#)
  - finite volume, [P-27](#)
  - generation, [U-136](#), [U-145](#)
  - grading, [U-136](#), [U-140](#)
  - grading, example of, [P-49](#)
  - non-orthogonal, [P-41](#)
  - refinement, [P-58](#)
  - resolution, [U-30](#)
  - specification, [U-127](#)
  - split-hex, [U-145](#)
  - Stereolithography (STL), [U-145](#)
  - surface, [U-145](#)
  - validity constraints, [U-127](#)
- Mesh Parts window panel, [U-23](#)
- meshes tools, [U-96](#)
- meshQualityControls keyword, [U-146](#)
- meshTools
  - library, [U-97](#)
- message passing interface
  - openMPI, [U-81](#)
- method keyword, [U-81](#)
- metis
  - keyword entry, [U-81](#)
- metisDecomp model, [U-97](#)
- MGridGenGAMGAgglomeration
  - library, [U-97](#)
- MGridGen
  - keyword entry, [U-123](#)
- mhdFoam solver, [P-65](#), [U-88](#)
- midPoint
  - keyword entry, [U-116](#)
- midPoint keyword, [U-175](#)
- midPointAndFace keyword, [U-175](#)
- min
  - tensor member function, [P-21](#)
- minArea keyword, [U-153](#)
- minDeterminant keyword, [U-153](#)
- minFaceWeight keyword, [U-153](#)

- minFlatness keyword, [U-153](#)
- minMedianAxisAngle keyword, [U-152](#)
- MINMOD differencing, [P-34](#)
- minRefinementCells keyword, [U-148](#)
- minThickness keyword, [U-152](#)
- minTriangleTwist keyword, [U-153](#)
- minTwist keyword, [U-153](#)
- minVol keyword, [U-153](#)
- minVolRatio keyword, [U-153](#)
- mirrorMesh utility, [U-90](#)
- mixed
  - boundary condition, [U-136](#)
- mixedSmagorinsky model, [U-101](#)
- mixtureAdiabaticFlameT utility, [U-95](#)
- mode keyword, [U-149](#)
- model
  - APIfunctions, [U-99](#)
  - BirdCarreau, [U-101](#)
  - CrossPowerLaw, [U-101](#)
  - DeardorffDiffStress, [U-101](#)
  - GuldersEGRLaminarFlameSpeed, [U-98](#)
  - GuldersLaminarFlameSpeed, [U-98](#)
  - HerschelBulkley, [U-101](#)
  - LRDDiffStress, [U-101](#)
  - LRR, [U-100](#)
  - LamBremhorstKE, [U-100](#)
  - LaunderGibsonRSTM, [U-100](#)
  - LaunderSharmaKE, [U-100](#)
  - LienCubicKELowRe, [U-100](#)
  - LienCubicKE, [U-100](#)
  - LienLeschzinerLowRe, [U-100](#)
  - NSRDSfunctions, [U-99](#)
  - Newtonian, [U-101](#)
  - NonlinearKEShah, [U-100](#)
  - PrandtlDelta, [U-101](#)
  - RNGkEpsilon, [U-100](#)
  - RaviPetersen, [U-98](#)
  - Smagorinsky2, [U-101](#)
  - Smagorinsky, [U-101](#)
  - SpalartAllmarasDDES, [U-101](#)
  - SpalartAllmarasIDDES, [U-101](#)
  - SpalartAllmaras, [U-100](#), [U-101](#)
  - adiabaticPerfectFluid, [U-99](#), [U-179](#)
  - anisotropicFilter, [U-100](#)
  - basicMultiComponentMixture, [U-98](#), [U-180](#)
  - chemistryModel, [U-99](#)
  - chemistrySolver, [U-99](#)
  - constTransport, [U-99](#), [U-180](#)
  - constant, [U-98](#)
  - cubeRootVolDelta, [U-101](#)
  - decompose, [U-97](#)
  - distributed, [U-97](#)
  - dynLagrangian, [U-101](#)
  - dynOneEqEddy, [U-101](#)
  - eConstThermo, [U-99](#), [U-179](#)
  - egrMixture, [U-98](#), [U-180](#)
  - hConstThermo, [U-99](#), [U-179](#)
  - hPolynomialThermo, [U-99](#), [U-180](#)
  - hePsiThermo, [U-98](#), [U-180](#)
  - heRhoThermo, [U-98](#), [U-180](#)
  - heheupsiReactionThermo, [U-98](#), [U-180](#)
  - homogenousDynOneEqEddy, [U-101](#)
  - homogenousDynSmagorinsky, [U-101](#)
  - homogeneousMixture, [U-98](#), [U-180](#)
  - icoPolynomial, [U-99](#), [U-179](#)
  - incompressiblePerfectGas, [U-99](#), [U-179](#)
  - inhomogeneousMixture, [U-98](#), [U-180](#)
  - interfaceProperties, [U-102](#)
  - janafThermo, [U-99](#), [U-180](#)
  - kEpsilon, [U-100](#)
  - kOmegaSSTAS, [U-101](#)
  - kOmegaSST, [U-100](#)
  - kOmega, [U-100](#)
  - kkLOmega, [U-100](#)
  - laminar, [U-100](#), [U-101](#)
  - laplaceFilter, [U-100](#)
  - lowReOneEqEddy, [U-101](#)
  - maxDeltaxyz, [U-101](#)
  - metisDecomp, [U-97](#)
  - mixedSmagorinsky, [U-101](#)
  - multiComponentMixture, [U-98](#), [U-180](#)
  - oneEqEddy, [U-101](#)
  - perfectFluid, [U-99](#), [U-179](#)
  - polynomialTransport, [U-99](#), [U-180](#)
  - powerLaw, [U-101](#)

- psiReactionThermo, [U-98](#), [U-180](#)
- psiuReactionThermo, [U-98](#), [U-180](#)
- ptsotchDecomp, [U-97](#)
- pureMixture, [U-98](#), [U-180](#)
- qZeta, [U-100](#)
- reactingMixture, [U-98](#), [U-180](#)
- realizableKE, [U-100](#)
- reconstruct, [U-97](#)
- rhoConst, [U-99](#), [U-179](#)
- rhoReactionThermo, [U-98](#), [U-180](#)
- scaleSimilarity, [U-101](#)
- scotchDecomp, [U-97](#)
- simpleFilter, [U-100](#)
- singleStepReactingMixture, [U-98](#), [U-180](#)
- smoothDelta, [U-101](#)
- specieThermo, [U-99](#), [U-180](#)
- spectEddyVisc, [U-101](#)
- sutherlandTransport, [U-99](#), [U-180](#)
- v2f, [U-100](#)
- vanDriestDelta, [U-101](#)
- veryInhomogeneousMixture, [U-98](#), [U-180](#)
- modifyMesh utility, [U-91](#)
- molecularMeasurements
  - library, [U-97](#)
- molecule
  - library, [U-97](#)
- molWeight keyword, [U-181](#)
- moveDynamicMesh utility, [U-90](#)
- moveEngineMesh utility, [U-90](#)
- moveMesh utility, [U-90](#)
- movingWallVelocity
  - boundary condition, [U-137](#)
- MPI
  - openMPI, [U-81](#)
- MRFInterFoam solver, [U-85](#)
- MRFMultiphaseInterFoam solver, [U-85](#)
- mshToFoam utility, [U-89](#)
- mu keyword, [U-182](#)
- multiComponentMixture model, [U-98](#), [U-180](#)
- multigrid
  - geometric-algebraic, [U-122](#)
- multiphaseEulerFoam solver, [U-85](#)
- multiphaseInterFoam solver, [U-86](#)
- MUSCL
  - keyword entry, [U-116](#)

## N

- n keyword, [U-81](#)
- nabla
  - operator, [P-23](#)
- nAlphaSubCycles keyword, [U-61](#)
- nBufferCellsNoExtrude keyword, [U-152](#)
- nCellsBetweenLevels keyword, [U-148](#)
- neighbour
  - dictionary, [U-129](#)
- neighbourPatch keyword, [U-142](#)
- netgenNeutralToFoam utility, [U-89](#)
- Newtonian
  - keyword entry, [U-58](#)
- Newtonian model, [U-101](#)
- nextWrite
  - keyword entry, [U-111](#)
- nFaces keyword, [U-130](#)
- nFinestSweeps keyword, [U-123](#)
- nGrow keyword, [U-152](#)
- nLayerIter keyword, [U-152](#)
- nMoles keyword, [U-181](#)
- non-orthogonal mesh, [P-41](#)
- nonBlocking
  - keyword entry, [U-78](#)
- none
  - keyword entry, [U-115](#), [U-122](#)
- NonlinearKEShik model, [U-100](#)
- nonNewtonianIcoFoam solver, [U-84](#)
- noWriteNow
  - keyword entry, [U-111](#)
- nPostSweeps keyword, [U-123](#)
- nPreSweeps keyword, [U-123](#)
- nRelaxedIter keyword, [U-152](#)
- nRelaxIter keyword, [U-150](#), [U-152](#)
- nSmoothNormals keyword, [U-152](#)
- nSmoothPatch keyword, [U-150](#)
- nSmoothScale keyword, [U-153](#)
- nSmoothSurfaceNormals keyword, [U-152](#)
- nSmoothThickness keyword, [U-152](#)

nSolveIter keyword, [U-150](#)  
NSRDSfunctions model, [U-99](#)  
null  
    keyword entry, [U-174](#)  
numberOfSubdomains keyword, [U-81](#)

## O

object keyword, [U-105](#)  
objToVTK utility, [U-90](#)  
ODE  
    library, [U-97](#)  
oneEqEddy model, [U-101](#)  
Opacity text box, [U-167](#)  
opaqueSolid  
    library, [U-98](#)  
OpenFOAM  
    applications, [U-67](#)  
    file format, [U-104](#)  
    libraries, [U-67](#)  
OpenFOAM  
    library, [U-96](#)  
OpenFOAM file syntax  
    //, [U-104](#)  
openMPI  
    message passing interface, [U-81](#)  
    MPI, [U-81](#)  
operator  
    scalar, [P-24](#)  
    vector, [P-23](#)  
Options window, [U-168](#)  
options file, [U-71](#)  
order keyword, [U-81](#)  
Orientation Axes button, [U-24](#), [U-167](#)  
orientFaceZone utility, [U-90](#)  
OSspecific  
    library, [U-97](#)  
outer product, **see** tensor, outer product  
outlet  
    boundary condition, [P-65](#)  
outletInlet  
    boundary condition, [U-137](#)  
outside  
    keyword entry, [U-149](#)  
owner

dictionary, [U-129](#)

## P

p field, [U-22](#)  
P1  
    library, [U-98](#)  
p rhgRefCell keyword, [U-124](#)  
p rhgRefValue keyword, [U-124](#)  
pairPatchAgglomeration  
    library, [U-97](#)  
paraFoam, [U-23](#), [U-163](#)  
parallel  
    running, [U-79](#)  
partialSlip  
    boundary condition, [U-137](#)  
particleTracks utility, [U-93](#)  
patch  
    boundary condition, [U-134](#)  
patch  
    keyword entry, [U-135](#), [U-176](#)  
patchAverage utility, [U-92](#)  
patches keyword, [U-138](#)  
patchIntegrate utility, [U-92](#)  
patchMap keyword, [U-160](#)  
patchSummary utility, [U-95](#)  
PBiCG  
    keyword entry, [U-121](#)  
PCG  
    keyword entry, [U-121](#)  
pdfPlot utility, [U-93](#)  
PDRFoam solver, [U-86](#)  
PDRMesh utility, [U-91](#)  
Pe utility, [U-92](#)  
perfectFluid model, [U-99](#), [U-179](#)  
permutation symbol, [P-15](#)  
pimpleDyMFoam solver, [U-84](#)  
pimpleFoam solver, [U-84](#)  
Pipeline Browser window, [U-23](#), [U-164](#)  
PISO  
    dictionary, [U-23](#)  
pisoFoam solver, [U-17](#), [U-84](#)  
Plot Over Line  
    menu entry, [U-34](#)  
plot3dToFoam utility, [U-90](#)



- pointField class, [P-27](#)
- pointField<Type> template class, [P-29](#)
- points
  - dictionary, [U-129](#), [U-136](#)
- polyBoundaryMesh class, [P-27](#)
- polyDualMesh utility, [U-90](#)
- polyLine
  - keyword entry, [U-139](#)
- polyMesh directory, [U-104](#), [U-129](#)
- polyMesh class, [P-27](#), [U-127](#), [U-129](#)
- polynomialTransport model, [U-99](#), [U-180](#)
- polyPatch class, [P-27](#)
- polyPatchList class, [P-27](#)
- polySpline
  - keyword entry, [U-139](#)
- porousInterFoam solver, [U-86](#)
- porousSimpleFoam solver, [U-84](#)
- post-processing, [U-163](#)
  - post-processing
    - paraFoam, [U-163](#)
- postCalc
  - library, [U-96](#)
- postChannel utility, [U-93](#)
- potentialFreeSurfaceFoam solver, [U-86](#)
- potential
  - library, [U-97](#)
- potentialFoam solver, [P-42](#), [U-83](#)
- pow
  - tensor member function, [P-21](#)
- powerLaw model, [U-101](#)
- pPrime2 utility, [U-92](#)
- Pr keyword, [U-182](#)
- PrandtlDelta model, [U-101](#)
- preconditioner keyword, [U-121](#), [U-122](#)
- pRefCell keyword, [U-23](#), [U-124](#)
- pRefValue keyword, [U-23](#), [U-124](#)
- pressure keyword, [U-49](#)
- pressure waves
  - in liquids, [P-58](#)
- pressureDirectedInletVelocity
  - boundary condition, [U-137](#)
- pressureInletVelocity
  - boundary condition, [U-137](#)
- pressureOutlet

- boundary condition, [P-59](#)
- pressureTransmissive
  - boundary condition, [U-137](#)
- primitive
  - library, [P-19](#)
- primitives tools, [U-96](#)
- printCoeffs keyword, [U-41](#), [U-184](#)
- processorWeights keyword, [U-80](#)
- probeLocations utility, [U-93](#)
- process
  - background, [U-24](#), [U-79](#)
  - foreground, [U-24](#)
- processor
  - boundary condition, [U-135](#)
- processor
  - keyword entry, [U-135](#)
- processorN directory, [U-80](#)
- processorWeights keyword, [U-81](#)
- Properties window panel, [U-25](#), [U-164](#)
- psiReactionThermo model, [U-98](#), [U-180](#)
- psiuReactionThermo model, [U-98](#), [U-180](#)
- ptot utility, [U-93](#)
- ptsotchDecomp model, [U-97](#)
- pureMixture model, [U-98](#), [U-180](#)
- purgeWrite keyword, [U-112](#)
- PV3FoamReader
  - library, [U-163](#)
- PVFoamReader
  - library, [U-163](#)

## Q

- Q utility, [U-92](#)
- QUICK
  - keyword entry, [U-119](#)
- qZeta model, [U-100](#)

## R

- R utility, [U-92](#)
- radiationModels
  - library, [U-98](#)
- randomProcesses
  - library, [U-97](#)
- RASModel
  - keyword entry, [U-40](#), [U-183](#)

RASModel keyword, [U-184](#)  
 RaviPetersen model, [U-98](#)  
 raw  
     keyword entry, [U-112](#), [U-174](#)  
 reactingFoam solver, [U-86](#)  
 reactingMixture model, [U-98](#), [U-180](#)  
 reactingParcelFilmFoam solver, [U-87](#)  
 reactingParcelFoam solver, [U-87](#)  
 reactionThermophysicalModels  
     library, [U-98](#)  
 realizableKE model, [U-100](#)  
 reconstruct model, [U-97](#)  
 reconstructPar utility, [U-83](#)  
 reconstructParMesh utility, [U-95](#)  
 redistributePar utility, [U-95](#)  
 refGradient keyword, [U-136](#)  
 refineHexMesh utility, [U-91](#)  
 refinementRegions keyword, [U-149](#)  
 refinementLevel utility, [U-91](#)  
 refinementRegions keyword, [U-148](#), [U-150](#)  
 refinementSurfaces keyword, [U-148](#)  
 refineMesh utility, [U-90](#)  
 refineWallLayer utility, [U-91](#)  
 Refresh Times button, [U-25](#)  
 regions keyword, [U-58](#)  
 relative tolerance, [U-121](#)  
 relativeSizes keyword, [U-152](#)  
 relaxed keyword, [U-153](#)  
 relTol keyword, [U-52](#), [U-121](#)  
 removeFaces utility, [U-91](#)  
 Render View window, [U-168](#)  
 Render View window panel, [U-168](#)  
 renumberMesh utility, [U-90](#)  
 Rescale to Data Range button, [U-25](#)  
 Reset button, [U-164](#)  
 resolveFeatureAngle keyword, [U-148](#)  
 restart, [U-38](#)  
 Reynolds number, [U-17](#), [U-21](#)  
 rhoPorousSimpleFoam solver, [U-84](#)  
 rhoReactingBuoyantFoam solver, [U-86](#)  
 rhoCentralDyMFoam solver, [U-84](#)  
 rhoCentralFoam solver, [U-84](#)  
 rhoConst model, [U-99](#), [U-179](#)  
 rhoLTSPimpleFoam solver, [U-84](#)

rhoPimpleFoam solver, [U-84](#)  
 rhoPimplecFoam solver, [U-84](#)  
 rhoReactingFoam solver, [U-86](#)  
 rhoReactionThermo model, [U-98](#), [U-180](#)  
 rhoSimpleFoam solver, [U-84](#)  
 rhoSimplecFoam solver, [U-84](#)  
 rmdepall script/alias, [U-74](#)  
 RNGkEpsilon model, [U-100](#)  
 roots keyword, [U-81](#), [U-82](#)  
 rotateMesh utility, [U-90](#)  
 run  
     parallel, [U-79](#)  
 run directory, [U-103](#)  
 runTime  
     keyword entry, [U-31](#), [U-111](#)  
 runTimeModifiable keyword, [U-112](#)

## S

sammToFoam utility, [U-90](#)  
 sample utility, [U-93](#), [U-173](#)  
 sampling  
     library, [U-96](#)  
 Save Animation  
     menu entry, [U-169](#)  
 Save Screenshot  
     menu entry, [U-169](#)  
 scalar, [P-12](#)  
     operator, [P-24](#)  
 scalar class, [P-19](#)  
 scalarField class, [P-25](#)  
 scalarTransportFoam solver, [U-83](#)  
 scale  
     tensor member function, [P-21](#)  
 scalePoints utility, [U-157](#)  
 scaleSimilarity model, [U-101](#)  
 scheduled  
     keyword entry, [U-78](#)  
 scientific  
     keyword entry, [U-112](#)  
 scotch  
     keyword entry, [U-80](#), [U-81](#)  
 scotchCoeffs keyword, [U-81](#)  
 scotchDecomp model, [U-97](#)  
 script/alias

- foamCorrectVrt, [U-158](#)
- foamJob, [U-177](#)
- foamLog, [U-177](#)
- make, [U-69](#)
- rmdepall, [U-74](#)
- wclean, [U-73](#)
- wmake, [U-69](#)
- second time derivative, [P-33](#)
- Seed window, [U-169](#)
- selectCells utility, [U-91](#)
- Set Ambient Color button, [U-166](#)
- setFields utility, [U-58](#), [U-89](#)
- setFormat keyword, [U-174](#)
- sets keyword, [U-174](#)
- setSet utility, [U-90](#)
- setsToZones utility, [U-90](#)
- Settings
  - menu entry, [U-168](#)
- settlingFoam solver, [U-86](#)
- SFCD
  - keyword entry, [U-116](#), [U-119](#)
- shallowWaterFoam solver, [U-84](#)
- shape, [U-140](#)
- Show Color Legend
  - menu entry, [U-25](#)
- SI units, [U-107](#)
- simpleReactingParcelFoam solver, [U-87](#)
- simple
  - keyword entry, [U-80](#), [U-81](#)
- simpleFilter model, [U-100](#)
- simpleFoam solver, [P-50](#), [U-84](#)
- simpleGrading keyword, [U-140](#)
- simpleSpline
  - keyword entry, [U-139](#)
- simulationType keyword, [U-40](#), [U-59](#), [U-183](#)
- singleCellMesh utility, [U-91](#)
- singleStepReactingMixture model, [U-98](#), [U-180](#)
- skew
  - tensor member function, [P-21](#)
- skewLinear
  - keyword entry, [U-116](#), [U-119](#)
- SLGThermo
  - library, [U-99](#)
- slice class, [P-27](#)
- slip
  - boundary condition, [U-137](#)
- Smagorinsky model, [U-101](#)
- Smagorinsky2 model, [U-101](#)
- snapToFoam utility, [U-91](#)
- smoothDelta model, [U-101](#)
- smoother keyword, [U-123](#)
- smoothSolver
  - keyword entry, [U-121](#)
- snap keyword, [U-146](#)
- snapControls keyword, [U-146](#)
- snappyHexMesh utility
  - background mesh, [U-146](#)
  - cell removal, [U-149](#)
  - cell splitting, [U-147](#)
  - mesh layers, [U-150](#)
  - meshing process, [U-145](#)
  - snapping to surfaces, [U-150](#)
- snappyHexMesh utility, [U-89](#), [U-145](#)
- snappyHexMeshDict file, [U-145](#)
- snGrad
  - fvc member function, [P-33](#)
- snGradCorrection
  - fvc member function, [P-33](#)
- snGradSchemes keyword, [U-114](#)
- Solid Color
  - menu entry, [U-166](#)
- solidChemistryModel
  - library, [U-99](#)
- solidDisplacementFoam solver, [U-88](#)
- solidDisplacementFoam solver, [U-50](#)
- solidEquilibriumDisplacementFoam solver, [U-88](#)
- solidMixtureProperties
  - library, [U-100](#)
- solidParticle
  - library, [U-97](#)
- solidProperties
  - library, [U-99](#)
- solidSpecie
  - library, [U-100](#)
- solidThermo

library, [U-100](#)

solver

DPMFoam, [U-87](#)

LTSInterFoam, [U-85](#)

LTSReactingFoam, [U-86](#)

LTSReactingParcelFoam, [U-87](#)

MRFInterFoam, [U-85](#)

MRFMultiphaseInterFoam, [U-85](#)

PDRFoam, [U-86](#)

SRFPimpleFoam, [U-84](#)

SRFSimpleFoam, [U-84](#)

XiFoam, [U-86](#)

adjointShapeOptimizationFoam, [U-83](#)

blockMesh, [P-43](#)

boundaryFoam, [U-84](#)

buoyantBoussinesqPimpleFoam, [U-86](#)

buoyantBoussinesqSimpleFoam, [U-86](#)

buoyantPimpleFoam, [U-86](#)

buoyantSimpleFoam, [U-86](#)

cavitatingDyMFoam, [U-85](#)

cavitatingFoam, [U-85](#)

chemFoam, [U-86](#)

chtMultiRegionFoam, [U-87](#)

chtMultiRegionSimpleFoam, [U-87](#)

coalChemistryFoam, [U-87](#)

coldEngineFoam, [U-86](#)

compressibleInterDyMFoam, [U-85](#)

compressibleInterFoam, [U-85](#)

compressibleMultiphaseInterFoam, [U-](#)

[85](#)

dnsFoam, [U-86](#)

dsmcFoam, [U-87](#)

electrostaticFoam, [U-88](#)

engineFoam, [U-86](#)

financialFoam, [U-88](#)

fireFoam, [U-86](#)

icoFoam, [U-17](#), [U-21](#), [U-22](#), [U-24](#), [U-](#)

[84](#)

icoUncoupledKinematicParcelDyMFoam,

[m,](#)

[U-87](#)

icoUncoupledKinematicParcelFoam,

[U-87](#)

interDyMFoam, [U-85](#)

interFoam, [U-85](#)

interMixingFoam, [U-85](#)

interPhaseChangeDyMFoam, [U-85](#)

interPhaseChangeFoam, [U-85](#)

laplacianFoam, [U-83](#)

magneticFoam, [U-88](#)

mdEquilibrationFoam, [U-87](#)

mdFoam, [U-87](#)

mhdFoam, [P-65](#), [U-88](#)

multiphaseEulerFoam, [U-85](#)

multiphaseInterFoam, [U-86](#)

nonNewtonianIcoFoam, [U-84](#)

pimpleDyMFoam, [U-84](#)

pimpleFoam, [U-84](#)

pisoFoam, [U-17](#), [U-84](#)

porousInterFoam, [U-86](#)

porousSimpleFoam, [U-84](#)

potentialFreeSurfaceFoam, [U-86](#)

potentialFoam, [P-42](#), [U-83](#)

reactingFoam, [U-86](#)

reactingParcelFilmFoam, [U-87](#)

reactingParcelFoam, [U-87](#)

rhoCentralDyMFoam, [U-84](#)

rhoCentralFoam, [U-84](#)

rhoLTSPimpleFoam, [U-84](#)

rhoPimpleFoam, [U-84](#)

rhoPimplecFoam, [U-84](#)

rhoReactingFoam, [U-86](#)

rhoSimpleFoam, [U-84](#)

rhoSimplecFoam, [U-84](#)

rhoPorousSimpleFoam, [U-84](#)

rhoReactingBuoyantFoam, [U-86](#)

scalarTransportFoam, [U-83](#)

settlingFoam, [U-86](#)

shallowWaterFoam, [U-84](#)

simpleReactingParcelFoam, [U-87](#)

simpleFoam, [P-50](#), [U-84](#)

solidDisplacementFoam, [U-88](#)

solidDisplacementFoam, [U-50](#)

solidEquilibriumDisplacementFoam,

[U-88](#)

sonicDyMFoam, [U-84](#)

sonicFoam, [P-56](#), [U-84](#)

sonicLiquidFoam, [P-59](#), [U-85](#)

- sprayEngineFoam, [U-87](#)
- sprayFoam, [U-87](#)
- thermoFoam, [U-87](#)
- twoLiquidMixingFoam, [U-86](#)
- twoPhaseEulerFoam, [U-86](#)
- uncoupledKinematicParcelFoam, [U-87](#)
- solver keyword, [U-52](#), [U-121](#)
- solver relative tolerance, [U-121](#)
- solver tolerance, [U-121](#)
- solvers keyword, [U-120](#)
- sonicDyMFoam solver, [U-84](#)
- sonicFoam solver, [P-56](#), [U-84](#)
- sonicLiquidFoam solver, [P-59](#), [U-85](#)
- source, [P-33](#)
- SpalartAllmaras model, [U-100](#), [U-101](#)
- SpalartAllmarasDDES model, [U-101](#)
- SpalartAllmarasIDDES model, [U-101](#)
- specie
  - library, [U-99](#)
- specie keyword, [U-181](#)
- specieThermo model, [U-99](#), [U-180](#)
- spectEddyVisc model, [U-101](#)
- spline keyword, [U-138](#)
- splitCells utility, [U-91](#)
- splitMesh utility, [U-91](#)
- splitMeshRegions utility, [U-91](#)
- spray
  - library, [U-97](#)
- sprayEngineFoam solver, [U-87](#)
- sprayFoam solver, [U-87](#)
- sqr
  - tensor member function, [P-21](#)
- sqrGradGrad
  - fvc member function, [P-33](#)
- SRFPimpleFoam solver, [U-84](#)
- SRFSimpleFoam solver, [U-84](#)
- star3ToFoam utility, [U-90](#)
- star4ToFoam utility, [U-90](#)
- startFace keyword, [U-130](#)
- startFrom keyword, [U-22](#), [U-111](#)
- starToFoam utility, [U-154](#)
- startTime
  - keyword entry, [U-22](#), [U-111](#)
- startTime keyword, [U-22](#), [U-111](#)
- steady flow
  - turbulent, [P-49](#)
- steadyParticleTracks utility, [U-93](#)
- steadyState
  - keyword entry, [U-119](#)
- Stereolithography (STL), [U-145](#)
- stitchMesh utility, [U-91](#)
- stl
  - keyword entry, [U-174](#)
- stopAt keyword, [U-111](#)
- strategy keyword, [U-80](#), [U-81](#)
- streamFunction utility, [U-92](#)
- stress analysis of plate with hole, [U-45](#)
- stressComponents utility, [U-92](#)
- Style window panel, [U-23](#), [U-166](#)
- Su
  - fvm member function, [P-33](#)
- subsetMesh utility, [U-91](#)
- summation convention, [P-13](#)
- SUPERBEE differencing, [P-34](#)
- supersonic flow, [P-55](#)
- supersonic flow over forward step, [P-54](#)
- supersonicFreeStream
  - boundary condition, [U-137](#)
- surfaceLambdaMuSmooth utility, [U-94](#)
- surface mesh, [U-145](#)
- surfaceAdd utility, [U-93](#)
- surfaceAutoPatch utility, [U-93](#)
- surfaceBooleanFeatures utility, [U-93](#)
- surfaceCheck utility, [U-93](#)
- surfaceClean utility, [U-93](#)
- surfaceCoarsen utility, [U-93](#)
- surfaceConvert utility, [U-93](#)
- surfaceFeatureConvert utility, [U-93](#)
- surfaceFeatureExtract utility, [U-93](#), [U-148](#)
- surfaceField<Type> template class, [P-29](#)
- surfaceFilmModels
  - library, [U-102](#)
- surfaceFind utility, [U-93](#)
- surfaceFormat keyword, [U-174](#)
- surfaceHookUp utility, [U-93](#)
- surfaceInertia utility, [U-94](#)
- surfaceMesh tools, [U-96](#)
- surfaceMeshConvert utility, [U-94](#)

[surfaceMeshConvertTesting utility, U-94](#)  
[surfaceMeshExport utility, U-94](#)  
[surfaceMeshImport utility, U-94](#)  
[surfaceMeshInfo utility, U-94](#)  
[surfaceMeshTriangulate utility, U-94](#)  
[surfaceNormalFixedValue](#)  
     boundary condition, [U-137](#)  
[surfaceOrient utility, U-94](#)  
[surfacePointMerge utility, U-94](#)  
[surfaceRedistributePar utility, U-94](#)  
[surfaceRefineRedGreen utility, U-94](#)  
[surfaces keyword, U-174](#)  
[surfaceSplitByPatch utility, U-94](#)  
[surfaceSplitByTopology utility, U-94](#)  
[surfaceSplitNonManifolds utility, U-94](#)  
[surfaceSubset utility, U-94](#)  
[surfaceToPatch utility, U-94](#)  
[surfaceTransformPoints utility, U-94](#)  
[surfMesh](#)  
     library, [U-97](#)  
[SuSp](#)  
     fvm member function, [P-33](#)  
[sutherlandTransport model, U-99, U-180](#)  
[symm](#)  
     tensor member function, [P-21](#)  
[symmetryPlane](#)  
     boundary condition, [P-59, U-134](#)  
[symmetryPlane](#)  
     keyword entry, [U-135](#)  
[symmTensorField class, P-25](#)  
[symmTensorThirdField class, P-25](#)  
[system directory, P-46, U-104](#)  
[systemCall](#)  
     library, [U-96](#)

## T

[T\(\)](#)  
     tensor member function, [P-21](#)  
[Tcommon keyword, U-182](#)  
[template class](#)  
     [GeometricBoundaryField, P-28](#)  
     [fvMatrix, P-29](#)  
     [dimensioned<Type>, P-21](#)

[FieldField<Type>, P-28](#)  
[Field<Type>, P-25](#)  
[geometricField<Type>, P-28](#)  
[List<Type>, P-25](#)  
[pointField<Type>, P-29](#)  
[surfaceField<Type>, P-29](#)  
[volField<Type>, P-29](#)  
[temporal discretisation, P-38](#)  
     Crank Nicholson, [P-38](#)  
     Euler implicit, [P-38](#)  
     explicit, [P-38](#)  
     in OpenFOAM, [P-39](#)  
[temporalInterpolate utility, U-93](#)  
[tensor, P-11](#)  
     addition, [P-13](#)  
     algebraic operations, [P-13](#)  
     algebraic operations in OpenFOAM, [P-19](#)  
     antisymmetric, [see](#) tensor, skew  
     calculus, [P-23](#)  
     classes in OpenFOAM, [P-19](#)  
     cofactors, [P-18](#)  
     component average, [P-16](#)  
     component maximum, [P-16](#)  
     component minimum, [P-16](#)  
     determinant, [P-18](#)  
     deviatoric, [P-17](#)  
     diagonal, [P-17](#)  
     dimension, [P-12](#)  
     double inner product, [P-15](#)  
     geometric transformation, [P-16](#)  
     Hodge dual, [P-18](#)  
     hydrostatic, [P-17](#)  
     identities, [P-17](#)  
     identity, [P-16](#)  
     inner product, [P-14](#)  
     inverse, [P-18](#)  
     magnitude, [P-16](#)  
     magnitude squared, [P-16](#)  
     mathematics, [P-11](#)  
     notation, [P-13](#)  
     nth power, [P-16](#)  
     outer product, [P-15](#)  
     rank, [P-12](#)

- rank 3, [P-12](#)
- scalar division, [P-14](#)
- scalar multiplication, [P-13](#)
- scale function, [P-16](#)
- second rank, [P-12](#)
- skew, [P-17](#)
- square of, [P-16](#)
- subtraction, [P-13](#)
- symmetric, [P-17](#)
- symmetric rank 2, [P-12](#)
- symmetric rank 3, [P-12](#)
- trace, [P-17](#)
- transformation, [P-16](#)
- transpose, [P-12](#), [P-17](#)
- triple inner product, [P-15](#)
- vector cross product, [P-15](#)
- tensor class, [P-19](#)
- tensor member function
  - [\\*](#), [P-21](#)
  - [+](#), [P-21](#)
  - [-](#), [P-21](#)
  - [/](#), [P-21](#)
  - [&](#), [P-21](#)
  - [&&](#), [P-21](#)
  - [^](#), [P-21](#)
  - [cmptAv](#), [P-21](#)
  - [cofactors](#), [P-21](#)
  - [det](#), [P-21](#)
  - [dev](#), [P-21](#)
  - [diag](#), [P-21](#)
  - [I](#), [P-21](#)
  - [inv](#), [P-21](#)
  - [mag](#), [P-21](#)
  - [magSqr](#), [P-21](#)
  - [max](#), [P-21](#)
  - [min](#), [P-21](#)
  - [pow](#), [P-21](#)
  - [scale](#), [P-21](#)
  - [skew](#), [P-21](#)
  - [sqr](#), [P-21](#)
  - [symm](#), [P-21](#)
  - [T\(\)](#), [P-21](#)
  - [tr](#), [P-21](#)
  - [transform](#), [P-21](#)
- tensorField class, [P-25](#)
- tensorThirdField class, [P-25](#)
- tetgenToFoam utility, [U-90](#)
- text box
  - Opacity, [U-167](#)
- thermalProperties
  - dictionary, [U-50](#)
- thermodynamics keyword, [U-181](#)
- thermoFoam solver, [U-87](#)
- thermophysical
  - library, [U-179](#)
- thermophysicalFunctions
  - library, [U-99](#)
- thermophysicalProperties
  - dictionary, [U-179](#)
- thermoType keyword, [U-179](#)
- Thigh keyword, [U-182](#)
- time
  - control, [U-111](#)
- time derivative, [P-33](#)
  - first, [P-35](#)
  - second, [P-33](#), [P-35](#)
- time step, [U-22](#)
- timeFormat keyword, [U-112](#)
- timePrecision keyword, [U-112](#)
- timeScheme keyword, [U-114](#)
- timeStamp
  - keyword entry, [U-78](#)
- timeStampMaster
  - keyword entry, [U-78](#)
- timeStep
  - keyword entry, [U-22](#), [U-31](#), [U-111](#)
- Tlow keyword, [U-182](#)
- tolerance
  - solver, [U-121](#)
  - solver relative, [U-121](#)
- tolerance keyword, [U-52](#), [U-121](#), [U-150](#)
- Toolbars
  - menu entry, [U-167](#)
- tools
  - algorithms, [U-96](#)
  - cfdTools, [U-96](#)
  - containers, [U-96](#)
  - db, [U-96](#)

- dimensionSet, [U-96](#)
- dimensionedTypes, [U-96](#)
- fields, [U-96](#)
- finiteVolume, [U-96](#)
- fvMatrices, [U-96](#)
- fvMesh, [U-96](#)
- global, [U-96](#)
- graph, [U-96](#)
- interpolations, [U-96](#)
- interpolation, [U-96](#)
- matrices, [U-96](#)
- memory, [U-96](#)
- meshes, [U-96](#)
- primitives, [U-96](#)
- surfaceMesh, [U-96](#)
- volMesh, [U-96](#)
- topoChangerFvMesh
  - library, [U-97](#)
- topoSet utility, [U-91](#)
- topoSetSource keyword, [U-58](#)
- totalPressure
  - boundary condition, [U-137](#)
- tr
  - tensor member function, [P-21](#)
- trace, [see](#) tensor, trace
- traction keyword, [U-49](#)
- transform
  - tensor member function, [P-21](#)
- transformPoints utility, [U-91](#)
- transport keyword, [U-181](#)
- transportProperties
  - dictionary, [U-21](#), [U-38](#), [U-41](#)
- transportProperties file, [U-58](#)
- triple inner product, [P-15](#)
- triSurface
  - library, [U-97](#)
- Ts keyword, [U-182](#)
- turbulence
  - dissipation, [U-39](#)
  - kinetic energy, [U-39](#)
  - length scale, [U-40](#)
- turbulence
  - library, [U-97](#)
- turbulence keyword, [U-184](#)

- turbulence model
  - RAS, [U-39](#)
- turbulenceProperties
  - dictionary, [U-40](#), [U-59](#), [U-183](#)
- turbulent flow
  - steady, [P-49](#)
- turbulentInlet
  - boundary condition, [U-137](#)
- tutorials
  - breaking of a dam, [U-55](#)
  - lid-driven cavity flow, [U-17](#)
  - stress analysis of plate with hole, [U-45](#)
- tutorials directory, [P-41](#), [U-17](#)
- twoLiquidMixingFoam solver, [U-86](#)
- twoPhaseEulerFoam solver, [U-86](#)
- twoPhaseProperties
  - library, [U-102](#)
- type keyword, [U-132](#), [U-133](#)

## U

- U field, [U-22](#)
- Ucomponents utility, [P-66](#)
- UMIST
  - keyword entry, [U-115](#)
- uncompressed
  - keyword entry, [U-112](#)
- uncorrected
  - keyword entry, [U-117](#), [U-118](#)
- uncoupledKinematicParcelFoam solver, [U-87](#)
- uniform keyword, [U-175](#)
- units
  - base, [U-107](#)
  - of measurement, [P-21](#), [U-107](#)
    - S.I. base, [P-21](#)
  - SI, [U-107](#)
  - Système International, [U-107](#)
  - United States Customary System, [U-107](#)
  - USCS, [U-107](#)
- Update GUI button, [U-165](#)
- uprime utility, [U-92](#)
- upwind
  - keyword entry, [U-116](#), [U-119](#)



- upwind differencing, [P-34](#), [U-61](#)
- USCS units, [U-107](#)
- Use Parallel Projection button, [U-24](#)
- Use parallel projection button, [U-167](#)
- utility
  - Co, [U-92](#)
  - Lambda2, [U-92](#)
  - Mach, [U-92](#)
  - PDRMesh, [U-91](#)
  - Pe, [U-92](#)
  - Q, [U-92](#)
  - R, [U-92](#)
  - Ucomponents, [P-66](#)
  - adiabaticFlameT, [U-95](#)
  - ansysToFoam, [U-89](#)
  - applyBoundaryLayer, [U-88](#)
  - applyWallFunctionBoundaryCondition
- s,
  - [U-88](#)
  - attachMesh, [U-90](#)
  - autoPatch, [U-90](#)
  - autoRefineMesh, [U-91](#)
  - blockMesh, [U-37](#), [U-89](#), [U-136](#)
  - boxTurb, [U-88](#)
  - ccm26ToFoam, [U-89](#)
  - cfx4ToFoam, [U-89](#), [U-154](#)
  - changeDictionary, [U-88](#)
  - checkMesh, [U-90](#), [U-155](#)
  - chemkinToFoam, [U-95](#)
  - collapseEdges, [U-91](#)
  - combinePatchFaces, [U-91](#)
  - createBaffles, [U-90](#)
  - createPatch, [U-90](#)
  - createTurbulenceFields, [U-92](#)
  - createExternalCoupledPatchGeometry, [U-88](#)
  - datToFoam, [U-89](#)
  - decomposePar, [U-79](#), [U-80](#), [U-95](#)
  - deformedGeom, [U-90](#)
  - dsmcFieldsCalc, [U-93](#)
  - dsmcInitialise, [U-88](#)
  - engineCompRatio, [U-93](#)
  - engineSwirl, [U-88](#)
  - ensight74FoamExec, [U-172](#)
  - ensightFoamReader, [U-91](#)
  - enstrophy, [U-92](#)
  - equilibriumCO, [U-95](#)
  - equilibriumFlameT, [U-95](#)
  - execFlowFunctionObjects, [U-93](#)
  - expandDictionary, [U-95](#)
  - extrude2DMesh, [U-89](#)
  - extrudeMesh, [U-89](#)
  - extrudeToRegionMesh, [U-89](#)
  - faceAgglomerate, [U-88](#)
  - flattenMesh, [U-90](#)
  - flowType, [U-92](#)
  - fluent3DMeshToFoam, [U-89](#)
  - fluentMeshToFoam, [U-89](#), [U-154](#)
  - foamCalc, [U-33](#), [U-93](#)
  - foamDataToFluent, [U-91](#), [U-170](#)
  - foamDebugSwitches, [U-95](#)
  - foamFormatConvert, [U-95](#)
  - foamHelp, [U-95](#)
  - foamInfoExec, [U-95](#)
  - foamListTimes, [U-93](#)
  - foamMeshToFluent, [U-89](#), [U-170](#)
  - foamToEnightParts, [U-91](#)
  - foamToEnight, [U-91](#)
  - foamToGMV, [U-91](#)
  - foamToStarMesh, [U-89](#)
  - foamToSurface, [U-89](#)
  - foamToTecplot360, [U-91](#)
  - foamToVTK, [U-91](#)
  - foamUpgradeCyclics, [U-88](#)
  - foamUpgradeFvSolution, [U-88](#)
  - foamyHexMesh, [U-89](#)
  - foamyQuadMesh, [U-89](#)
  - foamyHexMeshBackgroundMesh, [U-89](#)
  - foamyHexMeshSurfaceSimplify, [U-89](#)
  - gambitToFoam, [U-89](#), [U-154](#)
  - gmshToFoam, [U-89](#)
  - ideasToFoam, [U-154](#)
  - ideasUnvToFoam, [U-89](#)
  - insideCells, [U-90](#)
  - kivaToFoam, [U-89](#)
  - mapFields, [U-30](#), [U-37](#), [U-41](#), [U-54](#), [U-89](#),

U-160

mdInitialise, [U-89](#)

mergeMeshes, [U-90](#)

mergeOrSplitBaffles, [U-90](#)

mirrorMesh, [U-90](#)

mixtureAdiabaticFlameT, [U-95](#)

modifyMesh, [U-91](#)

moveDynamicMesh, [U-90](#)

moveEngineMesh, [U-90](#)

moveMesh, [U-90](#)

mshToFoam, [U-89](#)

netgenNeutralToFoam, [U-89](#)

objToVTK, [U-90](#)

orientFaceZone, [U-90](#)

pPrime2, [U-92](#)

particleTracks, [U-93](#)

patchAverage, [U-92](#)

patchIntegrate, [U-92](#)

patchSummary, [U-95](#)

pdfPlot, [U-93](#)

plot3dToFoam, [U-90](#)

polyDualMesh, [U-90](#)

postChannel, [U-93](#)

probeLocations, [U-93](#)

ptot, [U-93](#)

reconstructParMesh, [U-95](#)

reconstructPar, [U-83](#)

redistributePar, [U-95](#)

refineHexMesh, [U-91](#)

refineMesh, [U-90](#)

refineWallLayer, [U-91](#)

refinementLevel, [U-91](#)

removeFaces, [U-91](#)

renumberMesh, [U-90](#)

rotateMesh, [U-90](#)

sammToFoam, [U-90](#)

sample, [U-93](#), [U-173](#)

scalePoints, [U-157](#)

selectCells, [U-91](#)

setFields, [U-58](#), [U-89](#)

setSet, [U-90](#)

setsToZones, [U-90](#)

singleCellMesh, [U-91](#)

smapToFoam, [U-91](#)

snappyHexMesh, [U-89](#), [U-145](#)

splitCells, [U-91](#)

splitMeshRegions, [U-91](#)

splitMesh, [U-91](#)

star3ToFoam, [U-90](#)

star4ToFoam, [U-90](#)

starToFoam, [U-154](#)

steadyParticleTracks, [U-93](#)

stitchMesh, [U-91](#)

streamFunction, [U-92](#)

stressComponents, [U-92](#)

subsetMesh, [U-91](#)

surfaceLambdaMuSmooth, [U-94](#)

surfaceAdd, [U-93](#)

surfaceAutoPatch, [U-93](#)

surfaceBooleanFeatures, [U-93](#)

surfaceCheck, [U-93](#)

surfaceClean, [U-93](#)

surfaceCoarsen, [U-93](#)

surfaceConvert, [U-93](#)

surfaceFeatureConvert, [U-93](#)

surfaceFeatureExtract, [U-93](#), [U-148](#)

surfaceFind, [U-93](#)

surfaceHookUp, [U-93](#)

surfaceInertia, [U-94](#)

surfaceMeshConvertTesting, [U-94](#)

surfaceMeshConvert, [U-94](#)

surfaceMeshExport, [U-94](#)

surfaceMeshImport, [U-94](#)

surfaceMeshInfo, [U-94](#)

surfaceMeshTriangulate, [U-94](#)

surfaceOrient, [U-94](#)

surfacePointMerge, [U-94](#)

surfaceRedistributePar, [U-94](#)

surfaceRefineRedGreen, [U-94](#)

surfaceSplitByPatch, [U-94](#)

surfaceSplitByTopology, [U-94](#)

surfaceSplitNonManifolds, [U-94](#)

surfaceSubset, [U-94](#)

surfaceToPatch, [U-94](#)

surfaceTransformPoints, [U-94](#)

temporalInterpolate, [U-93](#)

tetgenToFoam, [U-90](#)

topoSet, [U-91](#)

- transformPoints, [U-91](#)
- uprime, [U-92](#)
- viewFactorsGen, [U-89](#)
- vorticity, [U-92](#)
- vtkUnstructuredToFoam, [U-90](#)
- wallFunctionTable, [U-89](#)
- wallGradU, [U-92](#)
- wallHeatFlux, [U-92](#)
- wallShearStress, [U-92](#)
- wdot, [U-93](#)
- writeCellCentres, [U-93](#)
- writeMeshObj, [U-90](#)
- yPlusLES, [U-92](#)
- yPlusRAS, [U-92](#)
- zipUpMesh, [U-91](#)
- utilityFunctionObjects
  - library, [U-96](#)

## V

- v2f model, [U-100](#)
- value keyword, [U-21](#), [U-136](#)
- valueFraction keyword, [U-136](#)
- van Leer differencing, [P-34](#)
- vanDriestDelta model, [U-101](#)
- vanLeer
  - keyword entry, [U-116](#)
- VCR Controls menu, [U-25](#), [U-165](#)
- vector, [P-12](#)
  - operator, [P-23](#)
  - unit, [P-16](#)
- vector class, [P-19](#), [U-107](#)
- vector product, **see** tensor, vector cross product
- vectorField class, [P-25](#)
- version keyword, [U-105](#)
- vertices keyword, [U-20](#), [U-138](#), [U-139](#)
- veryInhomogeneousMixture model, [U-98](#), [U-180](#)
- View menu, [U-167](#)
- View Settings
  - menu entry, [U-24](#), [U-167](#)
- View Settings (Render View) window, [U-167](#)
- View Settings...

- menu entry, [U-24](#)
- viewFactor
  - library, [U-98](#)
- viewFactorsGen utility, [U-89](#)
- viscosity
  - kinematic, [U-21](#), [U-41](#)
- volField<Type> template class, [P-29](#)
- volMesh tools, [U-96](#)
- vorticity utility, [U-92](#)
- vtk
  - keyword entry, [U-174](#)
- vtkFoam
  - library, [U-163](#)
- vtkPV3Foam
  - library, [U-163](#)
- vtkUnstructuredToFoam utility, [U-90](#)

## W

- wall
  - boundary condition, [P-59](#), [P-65](#), [U-57](#), [U-134](#)
- wall
  - keyword entry, [U-135](#)
- wallFunctionTable utility, [U-89](#)
- wallGradU utility, [U-92](#)
- wallHeatFlux utility, [U-92](#)
- Wallis
  - library, [U-99](#)
- wallShearStress utility, [U-92](#)
- wclean script/alias, [U-73](#)
- wdot utility, [U-93](#)
- wedge
  - boundary condition, [U-130](#), [U-135](#), [U-144](#)
- wedge
  - keyword entry, [U-135](#)
- window
  - Color Legend, [U-27](#)
  - Options, [U-168](#)
  - Pipeline Browser, [U-23](#), [U-164](#)
  - Render View, [U-168](#)
  - Seed, [U-169](#)
  - View Settings (Render View), [U-167](#)
- window panel

- Animations, [U-168](#)
- Annotation, [U-24](#), [U-167](#)
- Charts, [U-168](#)
- Color Legend, [U-166](#)
- Color Scale, [U-166](#)
- Colors, [U-168](#)
- Display, [U-23](#), [U-25](#), [U-164](#), [U-165](#)
- General, [U-167](#), [U-168](#)
- Information, [U-164](#)
- Lights, [U-167](#)
- Mesh Parts, [U-23](#)
- Properties, [U-25](#), [U-164](#)
- Render View, [U-168](#)
- Style, [U-23](#), [U-166](#)
- Wireframe
  - menu entry, [U-166](#)
- WM ARCH
  - environment variable, [U-74](#)
- WM ARCH OPTION
  - environment variable, [U-74](#)
- WM COMPILE OPTION
  - environment variable, [U-74](#)
- WM COMPILER
  - environment variable, [U-74](#)
- WM COMPILER BIN
  - environment variable, [U-74](#)
- WM COMPILER DIR
  - environment variable, [U-74](#)
- WM COMPILER LIB
  - environment variable, [U-74](#)
- WM DIR
  - environment variable, [U-74](#)
- WM MPLIB
  - environment variable, [U-74](#)
- WM OPTIONS
  - environment variable, [U-74](#)
- WM PRECISION OPTION
  - environment variable, [U-74](#)
- WM PROJECT
  - environment variable, [U-74](#)
- WM PROJECT DIR
  - environment variable, [U-74](#)
- WM PROJECT INST DIR
  - environment variable, [U-74](#)

- WM PROJECT USER DIR
  - environment variable, [U-74](#)
- WM PROJECT VERSION
  - environment variable, [U-74](#)
- wmake
  - platforms, [U-70](#)
- wmake script/alias, [U-69](#)
- word class, [P-21](#), [P-27](#)
- writeCellCentres utility, [U-93](#)
- writeCompression keyword, [U-112](#)
- writeControl
  - keyword entry, [U-111](#)
- writeControl keyword, [U-22](#), [U-60](#), [U-111](#)
- writeFormat keyword, [U-54](#), [U-112](#)
- writeInterval keyword, [U-22](#), [U-31](#), [U-111](#)
- writeMeshObj utility, [U-90](#)
- writeNow
  - keyword entry, [U-111](#)
- writePrecision keyword, [U-112](#)

## X

- x
  - keyword entry, [U-175](#)
- XiFoam solver, [U-86](#)
- xmgr
  - keyword entry, [U-112](#), [U-174](#)
- xyz
  - keyword entry, [U-175](#)

## Y

- y
  - keyword entry, [U-175](#)
- yPlusLES utility, [U-92](#)
- yPlusRAS utility, [U-92](#)

## Z

- z
  - keyword entry, [U-175](#)
- zeroGradient
  - boundary condition, [U-136](#)
- zipUpMesh utility, [U-91](#)