

UNIVERSIDADE FEDERAL DO PAMPA

DANILO NOGUEIRA LEMES JUNIOR

**INFLUÊNCIA DA PARAMETRIZAÇÃO DE TURBULÊNCIA NA SIMULAÇÃO
NUMÉRICA DE PROPAGAÇÃO DE MICROBURSTS**

Alegrete

2020

DANILO NOGUEIRA LEMES JUNIOR

**INFLUÊNCIA DA PARAMETRIZAÇÃO DE TURBULÊNCIA NA SIMULAÇÃO
NUMÉRICA DE PROPAGAÇÃO DE MICROBURSTS**

Dissertação de Mestrado apresentada ao Programa de Mestrado Stricto sensu em Engenharia da Universidade Federal do Pampa (UNIPAMPA, RS), como requisito parcial para obtenção do grau de Mestre em Engenharia.

Orientador: Prof. Felipe Denardin Costa
Co-orientador: Prof. Giuliano Demarco

Alegrete

2020

DANILO NOGUEIRA LEMES JUNIOR

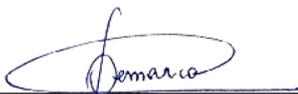
INFLUÊNCIA DA PARAMETRIZAÇÃO DE TURBULÊNCIA NA SIMULAÇÃO
NUMÉRICA DE PROPAGAÇÃO DE MICROBURSTS

Dissertação de Mestrado apresentada ao Programa de Mestrado Stricto sensu em Engenharia da Universidade Federal do Pampa (UNIPAMPA, RS), como requisito parcial para obtenção do grau de Mestre em Engenharia.

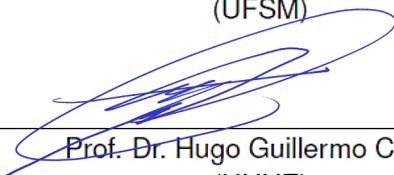
Dissertação de Mestrado defendida e aprovada em: 28 de dezembro de 2020.
Banca examinadora:



Prof. Dr. Felipe Denardin Costa
Orientador
(UNIPAMPA)



Prof. Dr. Giuliano Demarco,
Co-orientador
(UFSM)



Prof. Dr. Hugo Guillermo Castro
(UNNE)



Prof. Dr. Ernani de Lima Nascimento
(UFSM)

Aos meus pais.

RESUMO

O fenômeno *microburst* é caracterizado como uma forte corrente de ar descendente e localizada, produzindo ventos intensos altamente divergentes próximo à superfície, resultando em perfis de velocidade de vento distintos dos tipicamente observados na camada limite atmosférica. Tais características tornam o fenômeno um dos principais temas para estudos de análise de estruturas em engenharia. Diante disso, o objetivo inicial neste trabalho é a implementação do modelo de fonte de resfriamento introduzido por Anderson, Orf e Straka (1992) para análises de *microbursts* no software OpenFOAM. Após implementação do modelo proposto, observa-se que o mesmo apresenta as características do efluxo de um *microburst* real. Em um segundo momento, investiga-se a influência de diferentes parametrizações de turbulência na formação e propagação da frente de onda de um *microburst*. Os modelos de parametrização de turbulência utilizados consistem em dois modelos lineares e um não-linear, os quais fazem parte do conjunto de ferramentas do SOWFA (*Simulator fOr Wind Farm Applications*). Similaridades foram encontradas entre os modelos lineares, quanto ao desenvolvimento e difusividade na frente da onda, assim como a ocorrência das velocidades máximas em elevadas altitudes. Já o modelo não-linear apresentou menor difusividade na frente da onda durante a propagação, com as máximas velocidades concentrando-se em níveis menores do que os outros modelos. Diante disso é demonstrado que há influência das distintas parametrizações de turbulência na propagação de *microbursts*.

Palavras-chave: microburst; simulação dos grandes turbilhões; modelo de subgrade; SOWFA

ABSTRACT

The microburst phenomenon is characterized as a strongly localized downdraft, producing intense highly-divergent winds close to the surface, resulting in different wind speed profiles than those typically observed in the atmospheric boundary layer. Such characteristics make the phenomenon one of the main topics for studies of structural analysis in engineering. Therefore, the initial objective in this work is the implementation of the cooling source model introduced by Anderson, Orf e Straka (1992) for microbursts analysis in the OpenFOAM software. After implementing the proposed model, it is observed that it presents the outflow characteristics of a real microburst. In a second step, the influence of different turbulence parameterizations on the formation and propagation of the microburst wavefront is investigated. The turbulence parameterizations used consist of two linear and one nonlinear models, which are part of the SOWFA toolset. Similarities were found between linear models, regarding development and diffusivity in front of the wave, as well as the occurrence of maximum speeds at high altitudes. The nonlinear model showed lower diffusivity in front of the wave during propagation, with the maximum velocities concentrating at lower levels than the other models. Therefore, it is demonstrated that there is influence of the different turbulence parameterizations on the propagation of microbursts.

Keywords: microburst; large eddy simulation; sub-grid scale model; SOWFA

LISTA DE FIGURAS

- Figura 1 – Padrões mundiais da frequência de tempestades no planeta Terra. Representações das escalas de cores dadas em dias por ano. Ocorrências são mais frequentes na região entre os trópicos, onde a radiação do sol incide sobre a terra com maior intensidade, resultando em fortes correntes ascendentes. 15
- Figura 2 – Estágios do ciclo de vida de uma tempestade convectiva. (a) Estágio cúmulus. (b) Estágio maduro. (c) Estágio de dissipação. As linhas tracejadas em vermelho representam as isotermas das temperaturas 0°C e -40°C. . 16
- Figura 3 – Diferentes características de um *microburst* de acordo com a sua precipitação. *Microburst* úmido com a presença de forte chuva sobre a superfície, e *microburst* seco com pouca ou nenhuma presença de chuva sobre a superfície. 17
- Figura 4 – Vistas superiores do escoamento sobre a superfície de um *microburst* estático e não estático. *Microburst* estático forma a geometria de um anel anular simétrico, quando há ausência de ventos intensos sobre a superfície. *Microburst* não estático gera uma geometria elíptica. 18
- Figura 5 – Estágios de um *microburst* estático e não estático. (a) Esquema de um *microburst* estático, formando um colchão de ar frio sobre a superfície no decorrer do tempo. (b) Esquema de um *microburst* não estático, na presença de ventos intensos próximo à superfície, onde a camada de ar fria vai sendo deixada para trás conforme a nuvem avança. 18
- Figura 6 – Demonstração de um *microburst* no ar e de um *microburst* de superfície. O fenômeno somente é perceptível aos anemômetros quando alcança a superfície. 19
- Figura 7 – Evolução temporal do campo de velocidade de um *microburst*, baseado em alguns registros do projeto JAWS. A variável T representa o instante que a camada de ar começa divergir sobre o solo. 20
- Figura 8 – Característica do efluxo de um caso comum de *microburst* em diferentes estágios. 1° estágio: jato descendente forma um anel de vórtice. 2° estágio: contato com a superfície. 3° estágio: ventos intensos altamente divergentes são formados. 4° estágio: anel de vórtice chega em um limite, gerando diferentes rolos de vórtice. 20
- Figura 9 – Comparativo da velocidade do vento de um *microburst* registrado na Base da Força Aérea de Andrews (EUA) na altura de 4,9 m, com dados de ventos sobre uma zona rural na altura de 3 m. 22
- Figura 10 – Queda de aerogeradores no complexo eólico de Cerro Chato, RS, devido a um *microburst*. Ventos de quase 250 km h⁻¹ atingiram a região. 23

Figura 11 – Queda da ponte Morandi na cidade de Génova (Itália).	23
Figura 12 – Representação das escalas resolvidas em um escoamento turbulento para os modelos DNS, LES e RANS.	28
Figura 13 – Representação do perfil da velocidade junto a uma superfície sólida. Os dados do perfil de DNS são provenientes do trabalho de Lee e Moser (2015).	31
Figura 14 – Elevação da frente da camada de ar fria na simulação numérica de um <i>microburst</i> , devido a formação de um vórtice secundário próximo a superfície.	33
Figura 15 – Ilustração do volume da nuvem cúmulonimbus na simulação numérica. A região escura junto a nuvem que atinge o solo, representa a chuva da tempestade. Esse momento analisado é instantes antes dos ventos atingirem a máxima velocidade próximo da superfície.	34
Figura 16 – Evolução temporal do campo de velocidade de um <i>microburst</i> , utilizando o modelo de fonte de resfriamento. Os instantes analisados são: (a) $t = 300$ s, (b) $t = 338$ s, (c) $t = 370$ s e (d) $t = 500$ s.	35
Figura 17 – Velocidades máximas para diferentes rugosidades ao longo do tempo, utilizando o modelo de fonte de resfriamento. Os valores na legenda representam as rugosidades utilizadas no caso.	36
Figura 18 – Diferentes geometrias da fonte de resfriamento utilizadas nas simulações numéricas.	37
Figura 19 – Evolução temporal das velocidades radiais máximas, para diferentes características do termo fonte. A legenda é representada por $g(t)_{h_x}_{h_y}_{h_z}$	37
Figura 20 – Variação do forçante térmico ao longo do tempo, para máxima taxa de resfriamento de $-0,08 \text{ Ks}^{-1}$	39
Figura 21 – Variação espacial do termo fonte ao longo do raio da elipse, para diferentes expressões da função \cos^2	40
Figura 22 – Representação dos arquivos de cabeçalho e arquivo fonte do <i>solver sub-CloudPimpleFOAM</i>	50
Figura 23 – Domínio computacional utilizado no trabalho, com ilustração da fonte de resfriamento em azul.	54
Figura 24 – Campo da temperatura potencial dos diferentes modelos SGS, para o instante $t = 400$ s.	56
Figura 25 – Evolução temporal da diferença de temperatura potencial máxima e mínima, registrada em qualquer região do domínio computacional.	57
Figura 26 – Vista superior do plano da superfície, com a magnitude do campo da velocidade adimensional dos diferentes modelos SGS para o instante $t = 400$ s.	57

Figura 27 – Isosuperfícies do critério Q para valor de $Q = 0,0001$, e cortes transversais aproximadamente nos centros dos vórtices primários, apresentando o campo de velocidade adimensional para $t = 400$ s.	59
Figura 29 – Campo da vorticidade dos modelos SGS, para o instante $t = 320$ s. As Linhas de corrente e os <i>glyphs</i> são tangentes ao campo da velocidade. .	60
Figura 30 – Campo da vorticidade dos modelos SGS, para o instante $t = 360$ s. As Linhas de corrente e os <i>glyphs</i> são tangentes ao campo da velocidade. .	61
Figura 31 – Campo da vorticidade dos modelos SGS, para o instante $t = 400$ s. As Linhas de corrente e os <i>glyphs</i> são tangentes ao campo da velocidade. .	61
Figura 32 – Campo da vorticidade dos modelos SGS, para o instante $t = 430$ s. As Linhas de corrente e os <i>glyphs</i> são tangentes ao campo da velocidade. .	61
Figura 33 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 320$ s.	62
Figura 34 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 360$ s.	62
Figura 35 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 400$ s.	63
Figura 36 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 430$ s.	63
Figura 37 – Isosuperfícies do critério Q, para valor de $Q = 0,0001$, no instante $t = 430$ s.	64
Figura 38 – Evolução temporal dos picos máximos das velocidades nas simulações numéricas, registrados em qualquer região do plano diagonal do domínio computacional.	65
Figura 39 – Distâncias radiais e alturas das máximas velocidades em função do tempo.	66
Figura 40 – Evolução das máximas velocidades radiais em função do tempo, adimensionalizadas pelos picos radiais máximos dos ventos.	67
Figura 41 – Perfis da velocidade do vento na distância radial de 1500 m, nos instantes: 360 s, 400 s e 430 s.	68
Figura 42 – Perfis da velocidade do vento na distância radial de 2000 m, nos instantes: 360 s, 400 s e 430 s.	68
Figura 43 – Perfis adimensionais da velocidade radial do vento, próximo da região do pico máximo.	69

LISTA DE TABELAS

Tabela 1 – Classificações da estabilidade local atmosférica.	45
Tabela 2 – Esquemas numéricos utilizados no presente trabalho.	55

LISTA DE SIGLAS

<i>CFD</i>	Computational Fluid Dynamics
<i>DNS</i>	Direct Numerical Simulation
<i>JAWS</i>	Joint Airport Weather Studies
<i>LES</i>	Large Eddy Simulation
<i>MIST</i>	Microburst and Severe Thunderstorm
<i>NIMROD</i>	Northern Illinois Meteorological Research On Downburst
<i>NREL</i>	National Renewable Energy Laboratory
<i>OpenFOAM</i>	Open Field Operation and Manipulation
<i>PISO</i>	Pressure-Implicit With Splitting of Operators
<i>RANS</i>	Reynolds Averaged Navier-Stokes
<i>SOWFA</i>	Simulator fOr Wind Farm Applications
<i>SGS</i>	Sub-Grid Scale
<i>SIMPLE</i>	Semi-Implicit Method for Pressure-Linked Equations
<i>TASS</i>	Terminal Area Simulation System
<i>URANS</i>	Unsteady Reynolds averaged Navier Stokes

SUMÁRIO

1	INTRODUÇÃO	13
2	REVISÃO TEÓRICA.....	15
2.1	Estrutura e evolução de um <i>microburst</i>	15
2.2	Correntes de gravidade	21
2.3	Importância dos ventos de um <i>microburst</i> na engenharia	21
2.4	Modelos matemáticos para estudos numéricos de microbursts	24
2.4.1	Aproximação de Boussinesq	24
2.4.2	Campo da vorticidade	26
2.4.3	Modelo de turbulência	26
2.4.4	Simulação dos Grandes Turbilhões.....	28
2.4.5	Camada limite sobre placa plana	30
2.4.6	Modelagem numérica da ocorrência de <i>microbursts</i>	32
3	METODOLOGIA.....	38
3.1	Equações governantes	38
3.2	Modelos de subgrade	40
3.2.1	Modelo SmagorinskyABL	41
3.2.2	Modelo oneEqEddyABL.....	42
3.2.3	Modelo KosovicOneEqNBA	43
3.3	Comprimento característico de Pr_t e de subgrade	44
3.4	Modelo de parede	46
3.5	Implementação da fonte de resfriamento	47
3.6	Solver.....	49
3.7	Condições iniciais e de contorno	53
4	RESULTADOS E DISCUSSÕES.....	56
5	CONSIDERAÇÕES FINAIS.....	70
	REFERÊNCIAS BIBLIOGRÁFICAS	73
	Apêndice A – Condições iniciais de contorno	81
A.1	θ/U	81
A.2	θ/T	82
A.3	θ/p_{rgh}	83
A.4	$\theta/\nu Sgs$	84
A.5	$\theta/kappat$	85
A.6	θ/R_{wall}	86
A.7	θ/q_{wall}	87
A.8	θ/k	88
	Apêndice B – Modelo de turbulência e propriedades físicas	89

B.1	constant/LESProperties	89
B.2	constant/transportProperties	91
B.3	setup	92
B.4	constant/g	93
	Apêndice C – Soluções numéricas	94
C.1	system/blockMeshDict	94
C.2	system/controlDict	96
C.3	system/fvOptions	98
C.4	system/fvSchemes	100
C.5	system/fvSolution	101
	Apêndice D – Solucionador subCloudPimpleFoam	103
D.1	subCloudPimpleFoam.C	103
D.2	createFields.H	105
D.3	UEqn.H	108
D.4	computeBuoyancyTerm.H	109
D.5	TEqn.H	110
D.6	pEqn.H	111
D.7	turbulenceCorrect.H	113
D.8	readTransportProperties.H	114

1 INTRODUÇÃO

O termo *downburst* foi introduzido por Fujita e Byers (1977), através do seu estudo sobre as causas da queda de um avião em um aeroporto em Nova York, no ano de 1975. O evento foi ocasionado devido a mudanças abruptas e incomuns do vento. Esse fenômeno acontece em tempestades convectivas de grande variedade (WILSON et al., 1984), sendo definido como uma forte corrente de ar descendente, localizada, produzindo ventos intensos, altamente divergentes, próximo ou sobre a superfície (FUJITA; BYERS, 1977). Todavia, nem todas correntes descendentes produzem intensidade de um *downburst* (ROBERTS; WILSON, 1989). Antes da sua definição, meteorologistas acreditavam que os ventos descendentes perdiam intensidade conforme deslocavam-se em direção ao solo, até se tornarem inofensivos na superfície (FUJITA, 1990). No ano de 1983, a Base da Força Aérea de Andrews (EUA) ao estar sobre a ação do fenômeno, registrou ventos com intensidade de 67 ms^{-1} (FUJITA, 1990). Diante disso, além da preocupação na aviação, há também preocupação em projetos da engenharia. Apesar de existirem ocorrências de *downbursts* no Brasil, esses fenômenos ainda são pouco explorados em território brasileiro (LIMA; LOREDO-SOUZA, 2015).

Dentre as classificações de um *downburst*, o fenômeno denomina-se como *microburst* para alcance dos ventos extremos de até 4 Km horizontalmente, com duração entre 2 minutos e 5 minutos, e *macrobursts* para distâncias maiores, com tempo de duração dos ventos extremos entre 5 minutos e 20 minutos (FUJITA; WAKIMOTO, 1983). Os *microbursts* possuem uma natureza com escala espacial pequena e curto tempo de vida, dessa maneira eles são muito difíceis de serem monitorados (FUJITA, 1990). Diante desse fator, diversos estudos foram realizados de maneira a compreender melhor o fenômeno no decorrer dos anos, onde realizaram-se estudos experimentais (YAO; LUNDGREN, 1996; SENGUPTA; SARKAR, 2008; ZHANG; SARKAR; HU, 2015; DEMARCO et al., 2013, entre outros), estudos analíticos (VICROY, 1991; CHAY; ALBERMANI; WILSON, 2006; ROMANIC; PARVU; HANGAN, 2016, entre outros) e numéricos (SRIVASTAVA, 1985; PROCTOR, 1988, 1989; ANABOR et al., 2011; ABD-ELAAL; MILLS; MA, 2018, entre muitos outros).

Com o crescente aumento do poder computacional no decorrer dos anos, cada vez mais a simulação numérica vem sendo uma excelente ferramenta para estudos de casos complexos, com possibilidade de diferentes configurações e análises, auxiliando na pesquisa de campo, assim como nas simulações experimentais. Existem diferentes estratégias para modelar a geração e o comportamento de um *microburst* numericamente. Dentre os modelos numéricos, o modelo de jato incidente possui maior facilidade de implementação, o qual baseia-se em impulsionar um jato descendente em direção a superfície, sendo amplamente utilizado, tanto em estudos numéricos (KIM; HANGAN, 2007; QU et al., 2009; LI et al., 2012, entre outros), como em experimentais (LANDRETH; ADRIAN, 1990; MASON; WOOD, 2005; DAS; GHOSH; SINHAMAHAPATRA, 2011, entre outros).

Há modelos mais complexos para análise do fenômeno, tais como os modelos de nuvem completa, que são capazes de modelar todo ciclo de vida da nuvem para reprodução do *microburst* (HJELMFELT et al., 1989; PROCTOR, 1988, 1989; PROCTOR; BOWLES, 1992; STRAKA; ANDERSON, 1993; ORF; KANTOR; SAVORY, 2012). Todavia, tais modelos demandam um custo computacional muito alto, sendo inviáveis em análises da engenharia (ORES KOVIC, 2016). Diante disso, Anderson, Orf e Straka (1992) desenvolveram um modelo que simula somente o efluxo do modelo de nuvem completa, sendo um modelo mais simples, conhecido como modelo de fonte de resfriamento, o qual simula os efeitos microfísicos que acontecem dentro da nuvem (ORF; ANDERSON; STRAKA, 1996; ORF; ANDERSON, 1999; MASON; FLETCHER; WOOD, 2010; ANABOR et al., 2011). Através de análises comparativas entre os modelos de jato incidente e de fonte de resfriamento, estudos mostram diferenças entre as suas características de escoamento próximo à superfície. As diferenças devem-se que o modelo de jato incidente impõe parâmetros de força não físicos ao caso, não conseguindo reproduzir com precisão os efeitos de flu tuabilidade (VERMEIRE; ORF; SAVORY, 2011a). Por outro lado, o modelo de fonte de resfriamento consegue reproduzir esses efeitos, aproximando-se mais da realidade, por produzir a vortacidade de acordo com o termo de produção baroclínico (VERMEIRE; ORF; SAVORY, 2011a; ZHANG; HU; SARKAR, 2013).

Dessa forma, o presente trabalho tem como objetivo inicial, a implementação do modelo de fonte de resfriamento introduzido por Anderson, Orf e Straka (1992) no software de fluidodinâmica computacional (do inglês “*Computational Fluid Dynamics*” - CFD), de código aberto, OpenFOAM. A partir da implementação do modelo, a segunda proposta do trabalho é investigar como modelos de subgrade, ou subfiltro (do inglês “*Sub-Grid Scale*” - SGS), afetam a propagação de *microbursts* utilizando simulação dos grandes turbilhões (do inglês “*Large Eddy Simulation*” - LES). Para tanto, será comparada a performance de três parametrizações de turbulência para LES, na descrição do efluxo do fenômeno *microburst*. É importante ainda destacar que, o desenvolvimento do código numérico para o modelo de fonte de resfriamento, permitirá a disponibilização do mesmo para que se torne uma ferramenta a ser utilizada em estudos do fenômeno no OpenFoam.

O presente trabalho está dividido da seguinte forma: O Capítulo 2 apresenta a revisão teórica, exhibe os conceitos básicos utilizados no desenvolvimento do estudo, e o fenômeno é contextualizado para um melhor entendimento do caso. São retratadas algumas características da sua estrutura e evolução observadas em dados de campo, além da importância do seu estudo na engenharia. Além disso, são discutidos estudos anteriores que utilizam o mesmo forçante térmico deste trabalho. Posteriormente, no Capítulo 4 os resultados adquiridos no trabalho são apresentados e discutidos. Além da apresentação da performance do modelo de efluxo implementado no CFD, as análises comparativas entre os modelos SGS são apresentadas. Finalmente, o Capítulo 5 são expostas as conclusões observadas nos resultados, bem como, algumas perspectivas futuras.

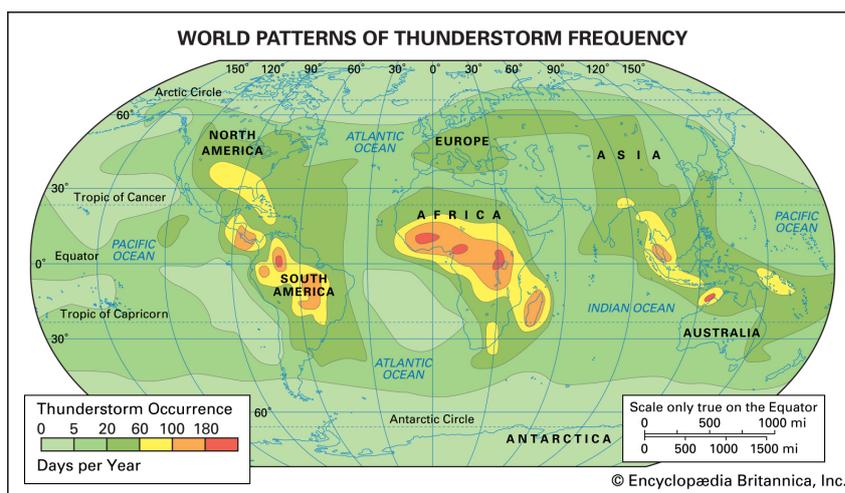
2 REVISÃO TEÓRICA

2.1 Estrutura e evolução de um *microburst*

Dentre os fatores para a formação de uma tempestade, está a presença de instabilidade atmosférica. Sobre tais circunstâncias, uma forte corrente de ar quente vertical pode ser formada, se uma quantidade de volume suficientemente grande deslocar-se para cima. Quando acompanhada de umidade, condensa-se e forma nuvens conhecidas como cúmulus, causando uma instabilidade ainda maior devido ao calor latente liberado nessa troca de estado, conseqüentemente intensificando o movimento ascendente do ar (KRIDER, 2016).

O mecanismo mais comum para a ascensão inicial da camada de ar, deve-se ao aquecimento da superfície terrestre pela radiação do sol. Com suficiente aquecimento, provoca intensas correntes de ar quente, devido a sua menor massa específica em relação ao ar frio. A instabilidade atmosférica também pode ocorrer em regiões montanhosas, as quais forçam os ventos a ascenderem, assim como em casos onde camadas de ar frio ganham energia ao transpassarem sobre a superfície de um oceano quente, ou sobre camadas de ar quente. Dessa maneira, a formação de tempestades são mais comuns nas regiões entre os trópicos, e nas regiões temperadas (Fig. 1) (KRIDER, 2016).

Figura 1 – Padrões mundiais da frequência de tempestades no planeta Terra. Representações das escalas de cores dadas em dias por ano. Ocorrências são mais frequentes na região entre os trópicos, onde a radiação do sol incide sobre a terra com maior intensidade, resultando em fortes correntes ascendentes.



Fonte: Adaptado de Krider (2016).

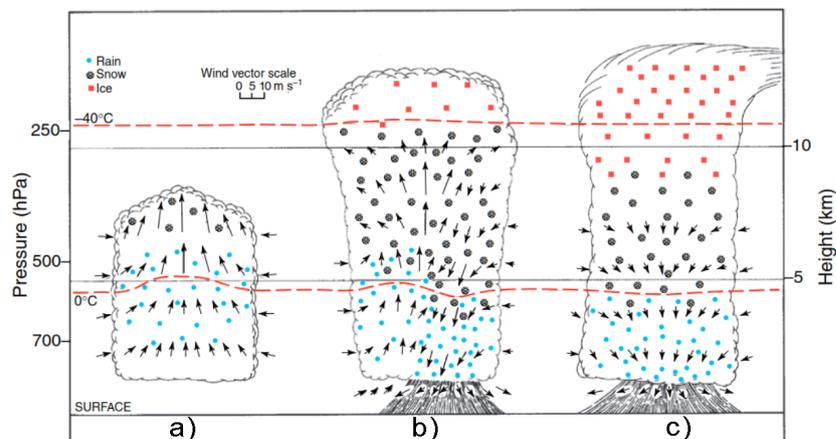
Toda tempestade em seu estágio inicial é uma nuvem cúmulus, porém, um pequeno número prossegue para todo o seu ciclo de desenvolvimento. Fatores como a instabilidade atmosférica e outras peculiaridades do ambiente são determinantes para esse avanço.

Byers e Braham (1949) detectaram três estágios no ciclo de vida de uma tempestade (Fig. 2). No primeiro estágio, conhecido como estágio cúmulus, há somente correntes ascendentes ao longo da nuvem, fazendo com que a nuvem cresça verticalmente. Nesse período inicial, a temperatura dentro da nuvem caracteriza-se como sendo mais quente que o ambiente a sua volta, atingindo velocidades máximas nas altitudes mais altas ao final do processo.

Conforme as linhas de corrente ascendem, há acúmulo de gotas e cristais de gelo na nuvem, devido ao aumento de condensação de vapor. Após um grande acúmulo, essas começam a cair, formando uma corrente em direção ao solo, dando início ao segundo estágio. Nesse estágio, conhecido como estágio maduro, existem tanto correntes ascendentes, como descendentes, onde uma área com baixa temperatura dentro da nuvem, coincide com a região das correntes descendentes.

No último estágio, conhecido como estágio de dissipação, tem seu início quando as correntes ascendentes cessam, havendo presença somente das correntes descendentes, onde com o tempo essas perdem intensidade, e então dissipam-se juntamente com a tempestade.

Figura 2 – Estágios do ciclo de vida de uma tempestade convectiva. (a) Estágio cúmulus. (b) Estágio maduro. (c) Estágio de dissipação. As linhas tracejadas em vermelho representam as isotermas das temperaturas 0°C e -40°C.



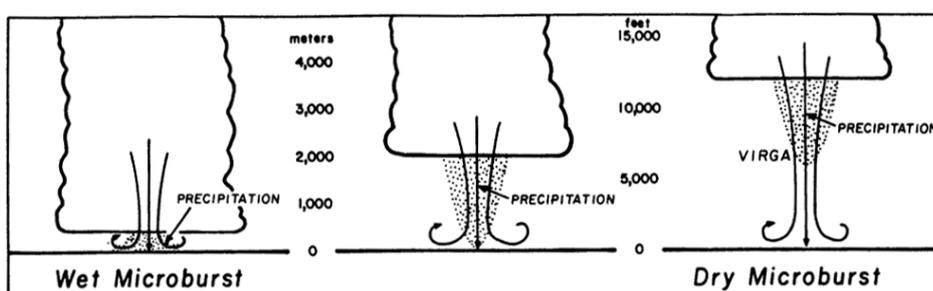
Fonte: Adaptado de Wallace e Hobbs (2006).

Microbursts ocorrem em tempestades severas, os quais podem ser classificados de acordo com a sua precipitação. Quando há pouca, ou nenhuma presença de chuva sobre a superfície, são denominados *microburst* seco. Já para os casos que o fenômeno é acompanhado por forte chuva, denomina-se *microburst* úmido (WAKIMOTO, 1985). Grandes pesquisas de campo foram realizadas após o termo *downburst* ter sido introduzido para maior caracterização do fenômeno. Dentre os estudos, o projeto *Northern Illinois Meteorological Research On Downburst* (NIMROD) (FUJITA, 1979), constatou que 36 % das

ocorrências de *microburst* registradas não haviam presença de chuva sobre a superfície. Esse projeto foi realizado no norte de Illinois, localizada na região centro-oeste dos EUA. Enquanto que, 83 % das ocorrências observadas no projeto *Joint Airport Weather Studies* (JAWS), não tinham a presença de chuva sobre a superfície. Essa observação já era esperada, pois esse projeto foi realizado próximo de Denver, Colorado, em altas planícies, sendo um terreno propício a formação de downburst seco (FUJITA; WAKIMOTO, 1983).

Na Fig. 3 são representadas três tipos de nuvens de *microburst* registradas nos projetos JAWS, NIMROD e no projeto *Microburst and Severe Thunderstorm* (MIST) (WAKIMOTO; BRINGI, 1988). Habitualmente, o *microburst* seco apresenta uma elevada altura da base da nuvem convectiva, ocorrendo em regiões secas. Já em regiões úmidas, é esperado que aconteça o *microburst* úmido, com menores elevações da altura da base da nuvem (FUJITA, 1985). O principal mecanismo de força do *microburst* seco é o resfriamento evaporativo da precipitação, já em *microbursts* úmidos, diferentes fatores regem o seu mecanismo de força (WAKIMOTO, 1985).

Figura 3 – Diferentes características de um *microburst* de acordo com a sua precipitação. *Microburst* úmido com a presença de forte chuva sobre a superfície, e *microburst* seco com pouca ou nenhuma presença de chuva sobre a superfície.



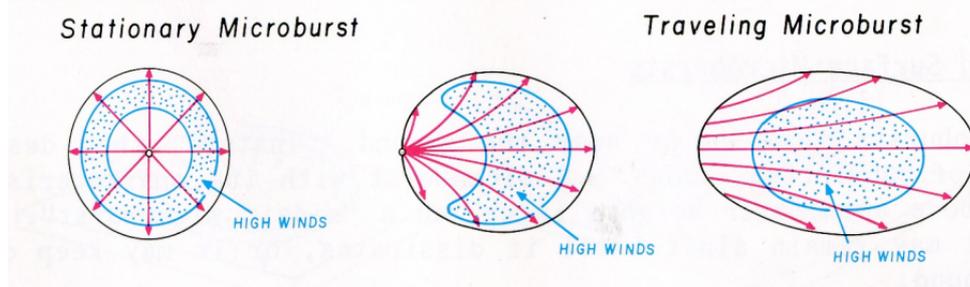
Fonte: Adaptado de Fujita (1985).

Diante das diversas características que um *microburst* pode ter, o fenômeno apresenta diferentes relações entre a altura e a largura da coluna descendente de ar. Para taxas de aspecto com menor largura e maior altura, ventos verticais mais intensos são formados. Esse atributo deve-se ao fato da camada descende do *microburst* ter que vencer a camada de ar abaixo, sendo dessa maneira mais eficiente com essa relação. Outros fatores também influenciam a intensidade dos ventos verticais. Srivastava (1985) ao realizar estudos com um modelo unidimensional, para análise de jato descendente conduzido por evaporação, observou diferentes fatores que aumentam a sua intensidade. Alguns dos fatores constatados que intensificam a corrente, são: aumento da taxa de lapso da temperatura e aumento da umidade relativa em um ambiente.

Na ausência de movimento da nuvem, classifica-se o fenômeno como *microburst* estático (tradução livre do inglês "*stationary microburst*"). Nesses casos, o escoamento

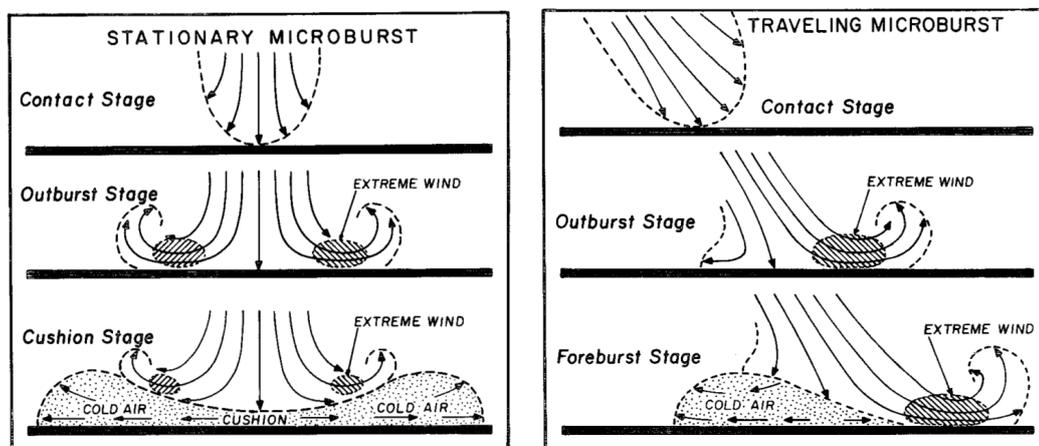
em contato com o solo de um ambiente não perturbado, forma um anel anular simétrico (FUJITA, 1985) (Fig. 4). Conforme a camada de ar descende, uma coluna de ar frio começa ser formada, impedindo que o jato alcance o solo, resultando em um aumento da altura das velocidades máximas com o decorrer do tempo (Fig. 5) (FUJITA, 1981). Já para os casos quando há movimento da nuvem, denomina-se o fenômeno como *microburst* não estático (tradução livre do inglês “*traveling microburst*”). O escoamento do *microburst* em contato com o solo nesses casos, forma uma geometria elíptica, gerando uma área crescente de ventos intensos (Fig. 4) (FUJITA, 1985). A medida que a nuvem avança no *microburst* não estático, a camada de ar fria vai sendo deixada para trás (Fig. 5) (FUJITA, 1981).

Figura 4 – Vistas superiores do escoamento sobre a superfície de um *microburst* estático e não estático. *Microburst* estático forma a geometria de um anel anular simétrico, quando há ausência de ventos intensos sobre a superfície. *Microburst* não estático gera uma geometria elíptica.



Fonte: Adaptado de Fujita (1985).

Figura 5 – Estágios de um *microburst* estático e não estático. (a) Esquema de um *microburst* estático, formando um colchão de ar frio sobre a superfície no decorrer do tempo. (b) Esquema de um *microburst* não estático, na presença de ventos intensos próximo à superfície, onde a camada de ar fria vai sendo deixada para trás conforme a nuvem avança.



(a)

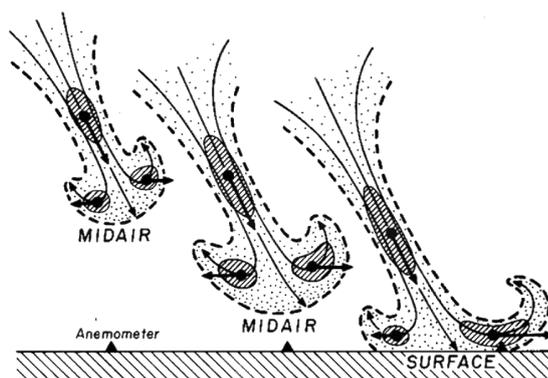
(b)

Fonte: Adaptado de Fujita (1981).

Conforme a camada de ar do *microburst* entra em contato com o vento cisalhante do ambiente, os ventos horizontais do fenômeno ajustam-se com esse escoamento, gerando perturbação no campo de velocidade do *microburst*. Diante disso, a translação do fenômeno está atrelada ao movimento da nuvem, enquanto a perturbação imposta em seu escoamento deve-se ao cisalhamento ambiente sob a nuvem (HJELMFELT, 1988).

Nem todos *microbursts* que descendem da base das nuvens alcançam a superfície. Para esses casos, que são imperceptíveis aos anemômetros terrestres, são classificados como *microbursts* no ar (tradução livre do inglês “*midair microbursts*”)(Fig. 6), os quais em um curto tempo podem descender e alcançar o solo, ou dissiparem-se no ar. Já os fenômenos que alcançam a superfície terrestre, são denominados *microbursts* de superfície (tradução livre do inglês “*surface microburst*”)(Fig. 6), onde as velocidades máximas geralmente encontram-se abaixo dos 100 m de altitude (FUJITA; WAKIMOTO, 1983).

Figura 6 – Demonstração de um *microburst* no ar e de um *microburst* de superfície. O fenômeno somente é perceptível aos anemômetros quando alcança a superfície.

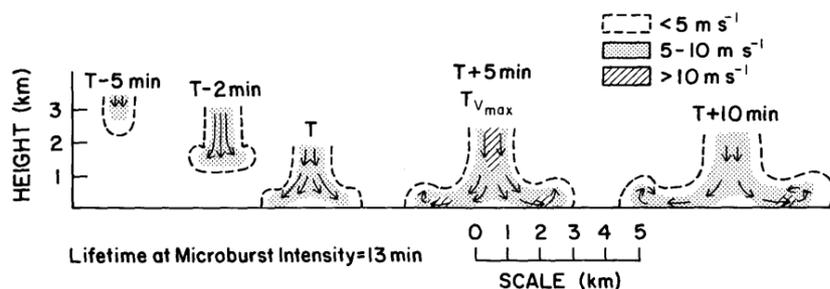


Fonte: Adaptado de Fujita e Wakimoto (1983).

Ao utilizar alguns casos do experimento JAWS, Wilson et al. (1984) descreveram a estrutura e evolução de um *microburst* baseado na média desses dados (Fig. 7). As relações são dadas em função da variável T , a qual representa o tempo que a camada de ar começa divergir no solo. No tempo $T - 5$ min, o escoamento do *microburst* encontra-se acima dos 2 km de altitude, representando a máxima taxa de precipitação do núcleo. Após 3 min, $T - 2$ min, o escoamento agora encontra-se abaixo de 1 km de altura, e então começa divergir horizontalmente, porém, com valores irrelevantes. Ao entrar em contato com o solo, no tempo T , uma zona de estagnação se forma, e o *microburst* começa desenvolver-se, aumentando o cisalhamento do vento e crescendo com o tempo, onde após 5 min, $T + 5$ min, o fenômeno alcança sua velocidade máxima. Com o passar do tempo, após $T + 10$ min, o fenômeno costuma perder intensidade, apresentando menor cisalhamento, assim como menores velocidades, tornando-se desorganizado (WILSON et al., 1984; HJELMFELT, 1988). Todavia, é importante destacar que esses valores são baseados em uma média, onde dada a complexidade do fenômeno, servem como uma

estimativa.

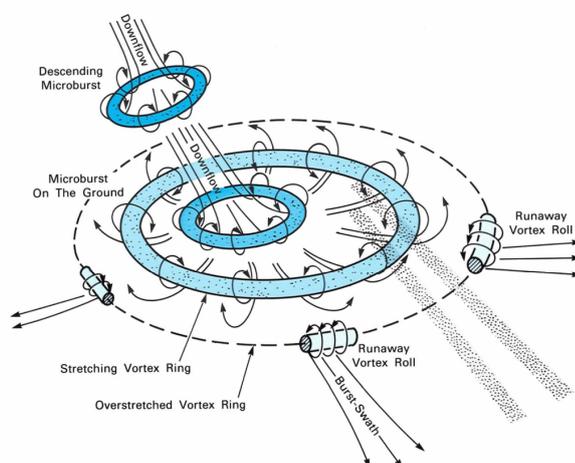
Figura 7 – Evolução temporal do campo de velocidade de um *microburst*, baseado em alguns registros do projeto JAWS. A variável T representa o instante que a camada de ar começa a divergir sobre o solo.



Fonte: Adaptado de Wilson et al. (1984).

A partir de um *microburst* registrado na Base da Força Aérea de Andrews (EUA), a Fig. 8 apresenta os diferentes estágios do fenômeno observados. Inicialmente, o jato descendente começa a formar um anel de vórtice conforme aproxima-se da superfície. Após o contato com o solo, esse vórtice começa a divergir radialmente (WOLFSON, 1988). Essa é uma característica frequente do fenômeno, porém, nem sempre esse aspecto é adquirido. Então o anel de vórtice expande-se conforme passa o tempo, até chegar em um limite, formando diferentes rolos de vórtices, cada um com um eixo horizontal, onde alguns desses rolos podem se distanciar para regiões distantes da sua origem, com tempo de duração de dois a três minutos (FUJITA, 1985).

Figura 8 – Característica do efluxo de um caso comum de *microburst* em diferentes estágios. 1º estágio: jato descendente forma um anel de vórtice. 2º estágio: contato com a superfície. 3º estágio: ventos intensos altamente divergentes são formados. 4º estágio: anel de vórtice chega em um limite, gerando diferentes rolos de vórtice.



Fonte: Adaptado de Wolfson (1988).

2.2 Correntes de gravidade

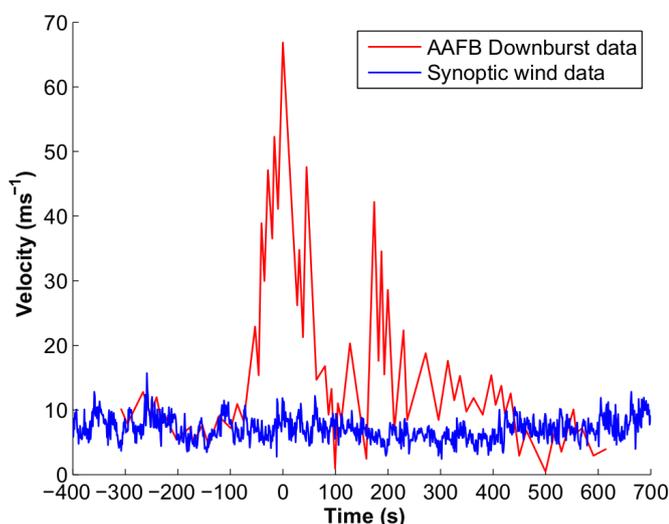
Quando fluídos com diferentes massas específicas escoam um sobre o outro, o escoamento formado nessa interação é denominado como correntes de gravidade, ou ainda, correntes de densidade (SIMPSON, 1982). A diferença de massa específica pode advir da diferença de material, ou diferença de temperatura, ou material dissolvido (SIMPSON, 1999). A formação de correntes de gravidade são observadas em diferentes situações naturais, assim como em casos por ação humana. Algumas das ocorrências na atmosfera, são: no efluxo de uma tempestade; e na frente de brisas marítimas. Assim como em diferentes situações nos oceanos, dentre as quais: induzidas por diferenças de salinidade e temperatura; e derramamento de óleo (SIMPSON, 1982).

Uma característica da propagação de correntes de gravidade é a formação de uma "cabeça" na borda frontal. Na maioria dos casos essa região é caracterizada por uma grande mistura, desempenhando um papel importante na propagação do escoamento (SIMPSON, 1982). Há duas principais formas de instabilidades que se formam nessa região. A primeira são instabilidades de Kelvin-Helmholtz, que desenvolvem-se sobre e atrás da cabeça da corrente de gravidade. A outra principal instabilidade observada são estruturas lóbulo e fenda, formadas por transbordamento do fluído menos denso na borda frontal (SIMPSON, 1999).

2.3 Importância dos ventos de um *microburst* na engenharia

Os *microbursts* atingem grandes intensidades de ventos, produzindo perfis de velocidade totalmente diferentes dos tipicamente presentes na camada limite atmosférica, sendo um fenômeno não estacionário. Projetos estruturais de velocidade de vento com períodos médios de retorno maiores que 10-20 anos, geralmente são atrelados a tempestades (SOLARI et al., 2015). Na Fig. 9 tem-se um comparativo da intensidade de ventos sinóticos em uma região rural na altura de 3 m, com ventos de um *microburst* registrado na Base da Força Aérea de Andrews (tradução livre do inglês "*Andrew's air force base*") na altura de 4,9 m (HAINES, 2015). Diante da sua complexidade, há dificuldades na formulação de um modelo fisicamente realista, existindo incertezas e problemas a serem resolvidos. Além disso, há limitação de dados disponíveis, devido ao seu curto tempo de duração, isso resulta em uma restrição tanto em modelos de estruturas adequados para cálculos de engenharia, como esquemas regulatórios (SOLARI, 2014).

Figura 9 – Comparativo da velocidade do vento de um *microburst* registrado na Base da Força Aérea de Andrews (EUA) na altura de 4,9 m, com dados de ventos sobre uma zona rural na altura de 3 m.



Fonte: Adaptado de Haines (2015).

Linhas de transmissão são propensas ao impacto de ventos extremos oriundos de *microbursts*, devido a estenderem-se por vários quilômetros (HOLMES, 2004). Estudos têm sido desenvolvidos ao longo dos anos para estudar a aerelasticidade desses objetos sobre ação do fenômeno (ABOSHOSHA; DAMATTY, 2015; ELAWADY et al., 2017; DAMATTY; ELAWADY, 2018). Dentre os desastres registrados devido a ventos intensos, está a queda de 19 torres de transmissão em Manitoba Hydro no ano de 1996, custando milhões de prejuízo a empresa (HYDRO, 2010).

A análise da carga de vento sobre aerogeradores é outro foco de estudos, devido a larga extensão territorial que os parques eólicos podem ocupar, estando dessa maneira sujeitos a ação do fenômeno (NGUYEN et al., 2013; NGUYEN; MANUEL, 2014; ZHANG; SARKAR; HU, 2015). No dia 20 de dezembro de 2014, em Santana do Livramento (RS), no complexo eólico de Cerro Chato, 8 aerogeradores dos 27 sofreram queda devido a um *microburst* (Fig. 10). Em nota de esclarecimento, a empresa Eletrosul informou que o fenômeno alcançou ventos com intensidade de quase 250 km h^{-1} , baseado em dados de institutos de meteorologia da região. A empresa salienta ainda que os aerogeradores tinham sido construídos de acordo com as normas, e não apresentavam problemas de construção diante dos testes e ensaios realizados (GONZATTO, 2016).

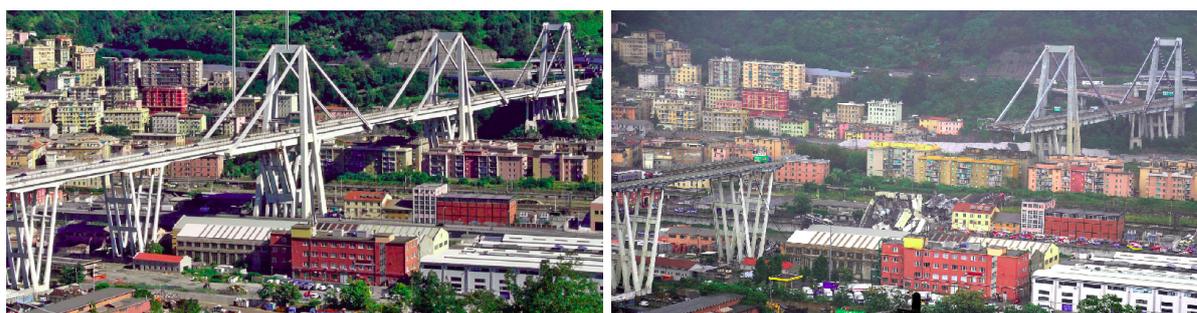
Figura 10 – Queda de aerogeradores no complexo eólico de Cerro Chato, RS, devido a um *microburst*. Ventos de quase 250 km h^{-1} atingiram a região.



Fonte: Adaptado de Gonzatto (2016).

Outro exemplo de danos em estruturas, que pode estar associado a um *microburst*, está a queda da ponte Morandi, localizada na cidade de Génova, Itália (Fig. 11). Investigações preliminares apontaram que no dia 14 de agosto de 2018 a queda teria sido provocada devido ao projeto inadequado, falta de manutenção constante e construção inadequada. Entretanto, Burlando et al. (2020) mostraram que, no momento da queda, a região da ponte estava sobre efeito de uma intensa tempestade. Porém, não há informações conclusivas, que o principal fator que influenciou a queda foram os fortes ventos, apesar de reconhecerem que esses podem ter influenciado na catástrofe.

Figura 11 – Queda da ponte Morandi na cidade de Génova (Itália).



Fonte: Adaptado de Pianigiani, Povoledo e Pérez-Peña (2018)

2.4 Modelos matemáticos para estudos numéricos de microbursts

2.4.1 Aproximação de Boussinesq

A aproximação de Boussinesq é bastante utilizada em análises da camada limite atmosférica, onde a dinâmica dos escoamentos é influenciada pelas diferenças de temperatura, resultando em grandes acelerações flutuantes (ARYA, 2001). Dentre as simplificações realizadas nas equações governamentais, o fluido é considerado como incompressível, e a variação da densidade é imposta somente onde é acoplada com a gravidade, sendo negligenciada nos outros termos (ZEYTOUNIAN, 1990). As equações de conservação de massa e de momento de um fluido incompressível, são dadas por (KUNDU; COHEN; DOWLING, 2008):

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (1a)$$

$$\rho \frac{\partial u_j}{\partial t} + \rho u_i \frac{\partial u_j}{\partial x_i} = - \frac{\partial p}{\partial x_j} + \rho g_j + \mu \frac{\partial^2 u_j}{\partial x_i^2} \quad (1b)$$

onde ρ e μ são a massa específica e viscosidade dinâmica do fluido, respectivamente.

Ao desconsiderar o termo $\rho^{-1}(D\rho/Dt)$ na equação de conservação de massa, a condição de incompressibilidade é utilizada em casos com baixa velocidade do escoamento, a qual pode ser relacionada pelo número de Mach (M). Essa variável adimensional é dada por, $M = U/c$, onde U e c representam as velocidades do escoamento e do som, respectivamente. Considera-se um fluido como incompressível para $M < 0,3$ (KUNDU; COHEN; DOWLING, 2008). Outra restrição na aproximação de Boussinesq é quanto a escala vertical do escoamento, a qual restringe-se para valores inferiores a $c^2 g^{-1} \sim 10$ Km. Para valores maiores que essa escala, as mudanças na densidade tornam-se importantes em um campo hidrostático (KUNDU; COHEN; DOWLING, 2008). Em casos tendo como fluido o ar, erros da ordem de 1 % são obtidos se a diferença de temperatura do caso é abaixo de 15 °C (FERZIGER; PERIĆ, 2002).

Quando considera-se um campo estacionário, assume-se um estado de referência atmosférico, o qual a partir da equação de momento (Eq. 1b), forma a seguinte equação hidrostática (ARYA, 2001):

$$0 = - \frac{\partial p_o}{\partial x_j} + \rho_o g_j \quad (2)$$

onde p_o é a pressão hidrostática, e ρ_o é a massa específica hidrostática.

Ao subtrair a Eq.2 da Eq.1b, e posteriormente dividir os termos por ρ_o , essas passam a ser dadas por:

$$\frac{\rho}{\rho_o} \frac{\partial u_j}{\partial t} + \frac{\rho}{\rho_o} u_i \frac{\partial u_j}{\partial x_i} = -\frac{1}{\rho_o} \frac{\partial p'}{\partial x_j} - \frac{\rho'}{\rho_o} g_j + \frac{\mu}{\rho_o} \frac{\partial^2 u_j}{\partial x_i^2} \quad (3)$$

a pressão passa a ser relacionada por $p' = p - p_o$, e a massa específica no termo da flutuabilidade é representado por $\rho' = \rho - \rho_o$. A aproximação de Boussinesq considera que há pequenas flutuações da densidade, dessa forma, considera-se $\rho/\rho_o \cong 1$ e $\mu/\rho_o \cong \nu$, onde ν é a viscosidade cinemática do fluído (KUNDU; COHEN; DOWLING, 2008). A equação de conservação de momento passa a ser expressa por:

$$\frac{\partial u_j}{\partial t} + u_i \frac{\partial u_j}{\partial x_i} = -\frac{1}{\rho_o} \frac{\partial p'}{\partial x_j} + \frac{\rho'}{\rho_o} g_j + \nu \frac{\partial^2 u_j}{\partial x_i^2} \quad (4)$$

A equação de energia de um fluído incompressível, dada em função da temperatura potencial (θ), negligenciando os termos da radiação e calor latente, é representada por (STULL, 1988):

$$\frac{\partial \theta}{\partial t} + u_i \frac{\partial \theta}{\partial x_i} = \alpha \frac{\partial^2 \theta}{\partial x_i^2} \quad (5)$$

onde α é a difusividade térmica, dada por $\alpha = \nu/P_r$, sendo P_r o número de Prandtl laminar.

A temperatura potencial é muito útil em análises meteorológicas, pois é uma propriedade conservativa para processos adiabáticos, sendo utilizada para diversas análises. Essa variável é denominada como a temperatura que uma parcela de ar não-saturado teria, se fosse transportada adiabaticamente em um processo de compressão ou expansão, de uma pressão e temperatura, p e T , até a pressão de 1000 hPa (ARYA, 2001). A sua relação é dada por:

$$\theta = T \left(\frac{1000}{p} \right)^{\frac{R}{c_p}} \quad (6)$$

onde R é a constante do gás, e c_p é o calor específico a pressão constante.

Em um fluído Newtoniano, o único modo reversível de trabalho é a compressão, onde as propriedades são dadas em função das variáveis termodinâmicas, pressão e temperatura (GRAY; GIORGINI, 1976). Entretanto, a aproximação de Boussinesq considera as propriedades do fluído como constantes (tais como c_p e ν), com exceção da massa específica (KUNDU; COHEN; DOWLING, 2008). A partir da equação, $\rho = \rho(T, P)$, uma expansão em série de Taylor é aplicada a essa, onde ao considerar o fluído como um gás ideal e realizar algumas simplificações, tais como a desconsideração dos efeitos da variação da pressão devido a contribuição da flutuação da temperatura ser muito mais significativa, passa a ser dada por (SPIEGEL; VERONIS, 1960):

$$\frac{(\rho - \rho_o)}{\rho_o} = -\frac{(T - T_o)}{T_o} \quad (7)$$

a massa específica passa a ser relacionada linearmente com a temperatura. Essa equação

também pode ser expressa de maneira similar com a temperatura potencial (STULL, 1988), realizando dessa forma o acoplamento entre as Eqs. 4 e 5.

2.4.2 Campo da vorticidade

A partir da rotação de elementos de fluídos, vórtices são formados (WU; MA; ZHOU, 2007), os quais são definidos como $\omega = \nabla \times u$ (DAVIDSON, 2015). Os vórtices são caracterizados por *streamlines* circulares, ou quase circulares (KUNDU; COHEN; DOWLING, 2008). Ao relacionar a sua definição acima com a equação de momento, considerando o movimento relativo da atmosfera devido a rotação da terra, a equação de transporte da vorticidade passa a ser dada por (WU; MA; ZHOU, 2007):

$$\frac{D\omega}{Dt} = \omega \cdot \nabla u + \frac{1}{\rho^2} \nabla \rho \times \nabla p + \nu \nabla^2 \omega \quad (8)$$

onde D/Dt é a derivada material, e a equação é dada em função da velocidade relativa e da vorticidade absoluta. O primeiro termo da esquerda da igualdade, representa a variação da vorticidade seguindo uma partícula de fluído; o primeiro termo do lado direito expressa o alongamento do vórtice; o segundo termo da direita é a taxa de geração de vorticidade devido a baroclinicidade do escoamento; e o último termo representa a taxa de mudança da vorticidade devido ao processo de difusão molecular (KUNDU; COHEN; DOWLING, 2008). O termo baroclínico é definido como um escoamento com duas ou mais variáveis termodinâmicas independentes. Um caso típico é a estratificação da densidade na atmosfera. Já para casos em que o escoamento possui somente uma variável termodinâmica independente, denomina-se escoamento barotrópico (WU; MA; ZHOU, 2007).

Kundu, Cohen e Dowling (2008) enfatizam que as três maneiras de gerar ou destruir um vórtice em um escoamento são: forças de corpo não conservativas; relação pressão-densidade não barotrópica; e forças viscosas diferentes de zero. Diante dessas considerações, mostra-se que a formação do anel de vórtice do *microburst* antes de tocar a superfície, está relacionado com o termo baroclínico. A coluna de ar descendente possui massa específica maior que o meio circundante, devido a sua baixa temperatura, com isso, a formação do vórtice deve-se principalmente ao gradiente horizontal da densidade, e também às instabilidades cisalhantes entre o *microburst* e o ambiente (VERMEIRE; ORF; SAVORY, 2011a).

2.4.3 Modelo de turbulência

Existem diferentes métodos numéricos para incluir os efeitos da turbulência em um escoamento, solucionando as equações de Navier-Stokes (DAVIDSON, 2015). A escolha do modelo a ser utilizado em uma análise depende de alguns fatores, tais como: nível de descrição da turbulência e custo computacional (POPE, 2001).

Dentre os modelos, a Simulação Numérica Direta (tradução livre do inglês “*Direct*”

Numerical Simulation” - DNS) resolve todas escalas de turbulência do escoamento, sem o uso de modelos de parametrização, discretizando as equações diretamente (Fig. 12) (DAVIDSON, 2015). Segundo a teoria da cascata de energia de Kolmogorov, supõe-se que os grandes turbilhões concentram a energia do escoamento, e a medida que esses turbilhões são divididos, a vorticidade se concentra em escalas cada vez menores, até que a energia do escoamento seja dissipada na forma de calor nas menores escalas do escoamento (POPE, 2001). Diante disso, o refinamento da malha computacional necessário para um caso DNS, deve ser suficientemente pequeno para resolver as escalas que são dissipadas na forma de calor. Outra necessidade é a de um passo de tempo suficientemente pequeno para resolver as flutuações mais rápidas (MALALASEKERA; VERSTEEG, 2007). O custo computacional para solucionar os casos é equivalente a $t \propto Re^3 Sc^2$, sendo $Sc = 1$ para os gases (ANDERSSON et al., 2011). Portanto, devido ao elevado custo computacional, a utilização de DNS é restrita a análises de escoamentos com baixo número de Reynolds e domínios geometricamente simples (ANDERSSON et al., 2011).

Uma alternativa viável a utilização de DNS é a Simulação dos Grandes Turbilhões, a qual resolve os maiores turbilhões, enquanto os menores são parametrizados (Fig. 12). Inicialmente o modelo LES realiza uma operação de filtragem nas equações de conservação de massa e momento, para decompor as variáveis em parte filtrada (ou resolvida), e residual (ou subgrade) (POPE, 2001):

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \quad (9a)$$

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial \overline{u_i u_j}}{\partial x_i} = -\frac{1}{\rho_o} \frac{\partial \bar{p}'}{\partial x_j} + \frac{(\rho - \rho_o)}{\rho_o} g_j + \frac{\partial}{\partial x_i} \left[\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] \quad (9b)$$

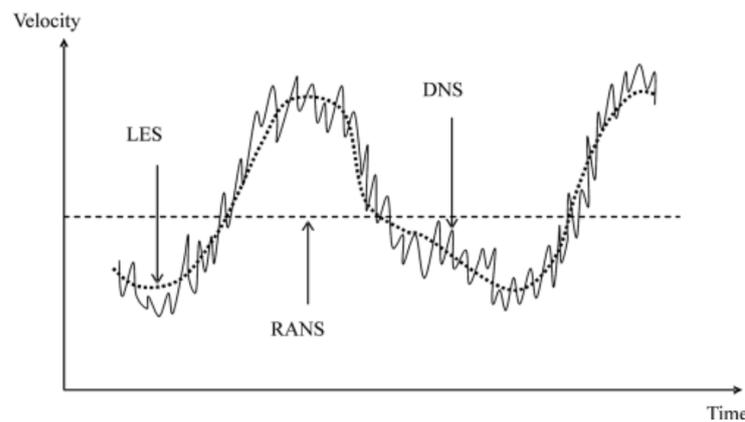
o termo $\partial \overline{u_i u_j} / \partial x_i$ não pode ser calculado, pois o sistema de equações possui mais variáveis desconhecidas do que equações, sendo dessa maneira uma variável desconhecida, criando um problema de fechamento. Contudo, através de modelos de turbulência é possível uma estimativa desse termo (HOLZMANN, 2016).

O modelo LES implica que pelo menos 80 % da energia cinética turbulenta do escoamento seja resolvida. Isso torna viável sua aplicação na solução de casos com maiores valores do número de Reynolds. Entretanto, o custo computacional ainda é muito elevado para determinados casos, como em aplicações em problemas de meteorologia. Isso deve-se a necessidade de uma grande resolução da malha nas regiões próximas as paredes. Todavia, esse problema pode ser contornado através do uso de uma parametrização para descrever o escoamento junto à parede, onde o modelo de turbulência resolverá 80 % da energia cinética turbulenta do escoamento, somente nas regiões mais distantes da superfície (POPE, 2001).

No outro extremo do DNS, está o modelo de Equações Médias de Reynolds (tradução livre do inglês “*Reynolds-averaged Navier–Stokes*” - RANS), que calcula apenas o

campo médio da velocidade em um escoamento estável (Fig. 12). Isso faz com que o modelo RANS seja muito utilizado na solução de problemas da engenharia, e em outros problemas mais simples (ANDERSSON et al., 2011). A partir do procedimento de decomposição de Reynolds, o escoamento é descrito estatisticamente através de variáveis médias e de flutuação, analogamente ao modelo LES. Contudo, as principais diferenças entre os modelos é que \bar{u} no modelo LES, não representa a média, mas a soma do movimento médio com as grandes escalas da turbulência, além do mais, possui as condições $\overline{u'} \neq 0$ e $\overline{\bar{u}} \neq \bar{u}$ (DAVIDSON, 2015). Para análises de escoamentos instáveis, há, ainda, o modelo de Equações Médias de Reynolds transiente (tradução livre do inglês “*unsteady Reynolds averaged Navier Stokes*” - URANS), que é um modelo RANS, com termos transientes incorporados na sua formulação.

Figura 12 – Representação das escalas resolvidas em um escoamento turbulento para os modelos DNS, LES e RANS.



Fonte: Adaptado de (ANDERSSON et al., 2011).

2.4.4 Simulação dos Grandes Turbilhões

Como descrito anteriormente, o modelo LES resolve as grandes escalas, enquanto as pequenas escalas são parametrizadas. A parametrização das pequenas escalas é conhecida como filtro de subgrade. Para se obter a equação de conservação de momento no modelo LES, é necessário algumas manipulações na Eq. 9b, a qual pode ser representada da seguinte maneira (DAVIDSON, 2015):

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial (\bar{u}_i \bar{u}_j)}{\partial x_i} = -\frac{1}{\rho_o} \frac{\partial \bar{p}'}{\partial x_j} + \frac{(\rho - \rho_o)}{\rho_o} g_j + \frac{\partial}{\partial x_i} \left[\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] - \frac{\partial \tau_{ij}^R}{\partial x_i} \quad (10)$$

onde $\tau_{ij}^R = \overline{u_i u_j} - \bar{u}_i \bar{u}_j$, sendo conhecido como tensor das tensões residuais (POPE,

2001), ou ainda, tensor das tensões em escala de subgrade (MALALASEKERA; VERSTEEG, 2007). Dessa maneira, ao decompor as velocidades desse tensor, tem-se a seguinte relação (MALALASEKERA; VERSTEEG, 2007):

$$\tau_{ij}^R = \overline{u_i u_j} - \bar{u}_i \bar{u}_j = (\overline{\bar{u}_i \bar{u}_j} - \bar{u}_i \bar{u}_j) + (\overline{\bar{u}_i u_j'} - \bar{u}_i' \bar{u}_j) + \overline{u_i' u_j'} \quad (11)$$

os primeiros termos no primeiro parênteses são conhecidos como tensões de Leonard, L_{ij} , os quais relacionam somente as escalas resolvidas; os termos no segundo parênteses são conhecidos como tensões cruzadas, C_{ij} , as quais estão atreladas a interação entre os turbilhões da escala de subgrade e o fluxo resolvido. O último termo em parênteses é denominado tensões de Reynolds, R_{ij} , sendo decorrida da troca de momento convectivo entre os vórtices não resolvidos presentes na SGS (MALALASEKERA; VERSTEEG, 2007; POPE, 2001).

O modelo LES realiza mais uma manipulação no tensor das tensões residuais para solucionar o problema de fechamento. O tensor é dividido em parte anisotrópica (Eq. 12a) e isotrópica, onde essa é adicionada em conjunto com o termo da pressão (Eq. 12b) (POPE, 2001):

$$\tau_{ij}^r = \tau_{ij}^R - \frac{2}{3} k_r \delta_{ij} \quad (12a)$$

$$\tilde{p} = \bar{p}' + \frac{2}{3} k_r \quad (12b)$$

onde k_r é a energia cinética residual, dada por $k_r = 1/2 \tau_{ii}^R$. Ao rearranjar esses termos na Eq. 10, essa toma a forma:

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial (\bar{u}_i \bar{u}_j)}{\partial x_i} = -\frac{1}{\rho_o} \frac{\partial \tilde{p}}{\partial x_j} + \frac{(\rho - \rho_o)}{\rho_o} g_j + \frac{\partial}{\partial x_i} \left[\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] - \frac{\partial \tau_{ij}^r}{\partial x_i} \quad (13)$$

Diferentes métodos foram desenvolvidos ao longo dos anos para solucionar o tensor das tensões residuais. Dentre esses, diversos modelos são relacionados pela hipótese da viscosidade turbulenta de Boussinesq, que impõe que as tensões de Reynolds poderiam ser relacionadas linearmente com a taxa média local de deformação (S_{ij}) (MALALASEKERA; VERSTEEG, 2007). Propondo, assim, que o transporte de momento pela turbulência é um processo difusivo, e o tensor das tensões residuais é relacionado somente pelas tensões de Reynolds (ANDERSSON et al., 2011). A característica da condição de linearidade é que ela força o alinhamento das tensões residuais com o tensor da taxa de deformação. No entanto, mesmo para condições de turbulência homogênea e isotrópica, existem estruturas em todas escalas na faixa inercial que não possuem isotropia (ABBÀ; CERCIGNANI; VALDETTARO, 2003).

A seguir, são demonstradas as manipulações necessárias na equação de energia

(Eq. 5) para análises utilizando LES. Ao realizar a operação de filtragem na equação de energia, tem-se:

$$\frac{\partial \bar{\theta}}{\partial t} + \frac{\partial \bar{\theta} \bar{u}_j}{\partial x_j} = \alpha \frac{\partial^2 \bar{\theta}}{\partial x_j^2} - \frac{\partial \tau_{\theta j}}{\partial x_j} \quad (14)$$

O termo $\tau_{\theta j}$ representa o fluxo turbulento de calor sensível de subgrade. As manipulações realizadas são semelhantes as do tensor das tensões residuais:

$$\tau_{\theta j} = \overline{u_j \theta} - \bar{u}_j \bar{\theta} \quad (15)$$

Ao decompor as variáveis do tensor em parte filtrada e residual, toma a seguinte forma (IRRENFRIED; STEINER, 2017):

$$\tau_{\theta j} = (\overline{\bar{u}_j \bar{\theta}} - \bar{u}_j \bar{\theta}) + (\overline{u'_j \bar{\theta}} + \overline{\bar{u}_j \theta'} - \bar{u}'_j \bar{\theta} - \bar{u}_j \bar{\theta}') + (\overline{u'_j \theta'} - \bar{u}'_j \bar{\theta}') \quad (16)$$

No lado direito da igualdade, os termos presentes no primeiro parênteses são os fluxos de calor de Leonard, que representam os fluxos de calor resolvidos; no segundo parênteses, os termos são conhecidos como fluxos de calor cruzados, que relacionam a interação entre os menores fluxos de calor resolvidos com os não resolvidos; e no último parênteses, têm-se os fluxos de calor de Reynolds, que representam a interação entre os fluxos de calor não resolvidos (IRRENFRIED; STEINER, 2017). Assim como o tensor das tensões residuais, o fluxo turbulento de calor sensível de subgrade é parametrizada por modelos SGS.

2.4.5 Camada limite sobre placa plana

O escoamento ao passar próximo da superfície, pode ser dividido em três regiões: subcamada viscosa (tradução livre do inglês “*sub-layer*”), camada de amortecimento (tradução livre do inglês “*buffer-layer*”) e região logarítmica (tradução livre do inglês “*log-region*”) (LIU, 2016). Alguns parâmetros são estabelecidos ao refinar a malha, de maneira que a altura da primeira célula do domínio computacional, esteja dentro da subcamada viscosa, para então conseguir reproduzir adequadamente os efeitos da camada limite.

A caracterização dessas regiões é dada em função de duas variáveis adimensionais. A primeira é a velocidade adimensional do escoamento:

$$u^+ = \frac{u}{u_\tau} \quad (17a)$$

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \quad (17b)$$

onde u_τ é a velocidade de fricção e τ_w é a tensão de cisalhamento da superfície.

A segunda variável adimensional, necessária para delimitação das regiões, é a dis-

tância não-dimensional em relação ao solo:

$$y^+ = \frac{yu_\tau}{\nu} \quad (18)$$

Na região da subcamada viscosa, há somente efeitos viscosos atuando sobre o fluido, assume-se, dessa maneira, que as tensões cisalhantes do fluido e da parede são iguais. Essa região localiza-se em $y^+ < 5$ (Fig. 13), sendo a velocidade adimensional relacionada por (LIU, 2016):

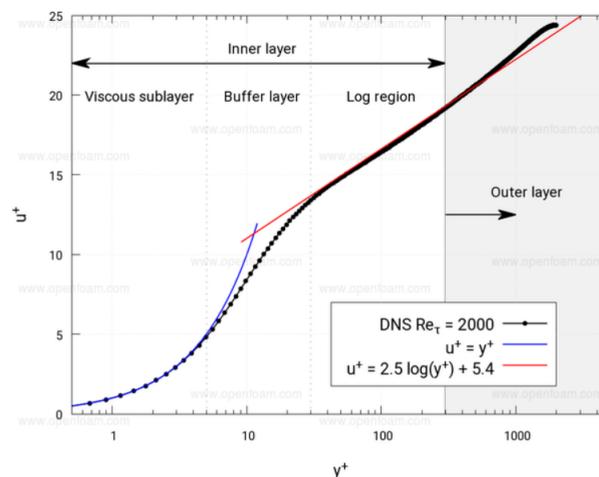
$$u^+ = y^+ \quad (19)$$

Na região logarítmica, localizada em $30 < y^+ < 200$ (Fig. 13), a tensão turbulenta prevalece sobre o escoamento, onde a velocidade adimensional é dada por (LIU, 2016):

$$u^+ = \frac{1}{\kappa} \ln(Ey^+) \quad (20)$$

onde E é uma constante empírica. A camada de amortecimento é localizada entre essas duas regiões (Fig. 13), onde os efeitos das tensões viscosas e turbulentas são de magnitudes semelhantes, possuindo um perfil complexo e difícil de ser modelado (LIU, 2016).

Figura 13 – Representação do perfil da velocidade junto a uma superfície sólida. Os dados do perfil de DNS são provenientes do trabalho de Lee e Moser (2015).



Fonte: Adaptado de (OPENCDF, 2020).

Essas três regiões compreendem a área conhecida como camada interna (tradução livre do inglês “*inner layer*”), onde há grande influência dos efeitos viscosos na parede. Após essa área, inicia-se a camada externa (tradução livre do inglês “*outer layer*”), onde os efeitos da viscosidade sobre a velocidade média são negligenciados.

O tempo computacional para solucionar a camada interna é muito maior do que a camada externa, onde necessita-se de um espaçamento constante da grade, devido a pre-

dominância de vórtices quasi-streamwise, cujas dimensões são constantes em unidades de parede. Dessa maneira, uma maneira de melhorar a performance em simulações que possuem alto valor do número de Reynolds, é parametrizar essa região e resolver apenas a camada externa (PIOMELLI; BALARAS, 2002).

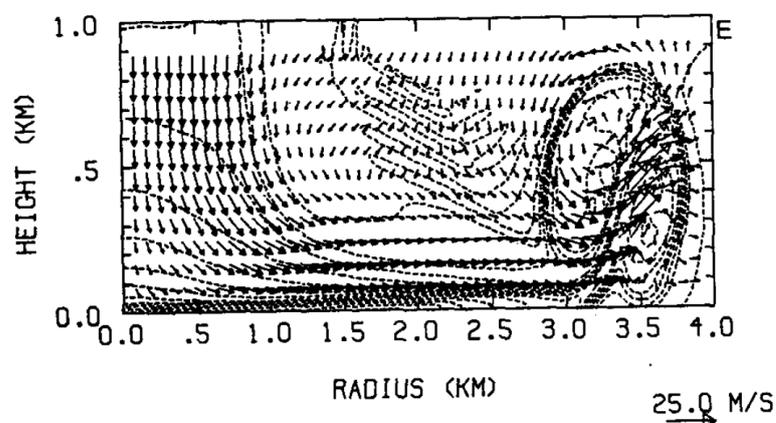
2.4.6 Modelagem numérica da ocorrência de *microbursts*

Diversos estudos numéricos foram realizados para analisar as características do *microburst* ao longo dos anos. Nesta seção serão destacados os resultados de alguns estudos, com uma breve descrição das metodologias utilizadas, para essas informações servirem como complemento e ajuda na compreensão dos resultados obtidos neste trabalho.

Nas análises desenvolvidas por Proctor (1988), o *microburst* foi dirigido por resfriamento evaporativo da chuva, tendo como condições do ambiente, valores registrados no projeto JAWS. O modelo numérico utilizado foi o *Terminal Area Simulation System* (TASS). O modelo possui doze equações prognósticas, sendo três para equação de momento, uma para pressão, uma para temperatura potencial, e mais seis equações prognósticas de conservação das substâncias da água, tais como: vapor de água, gotas de nuvens, cristais de gelo nas nuvens, chuva, neve e granizo (SWITZER; PROCTOR, 2014).

Proctor (1988) realizou a simulação em um domínio cilíndrico com 5 km de altura, por 10 km de largura, considerando o ambiente como seco-adiabático da superfície até a altura de 2 km. Ao simular o caso, o escoamento descendente apresenta um anel de vórtice antes de tocar o solo, que é propagado radialmente ao se chocar com a superfície, gerando picos de velocidade horizontais logo após alcançar a superfície. A velocidade de propagação diminui vagarosamente a sua intensidade, ao se propagar para distâncias maiores com o anel de vórtice. A velocidade máxima do escoamento de saída ocorre aproximadamente 4 min após a precipitação tocar o solo, onde a máxima velocidade horizontal acontece dentro de 80 m de altura. Outra observação importante é a influência da superfície sobre a estrutura da frente do anel de vórtice. Devido a sua fricção gerada na superfície, o anel de vórtice desenvolve uma contra-rotação próxima ao solo, formando uma rotação secundária, fazendo com que aumente a altura da frente da propagação, e posteriormente essa seja varrida para cima, levando ventos intensos para maiores altitudes (Fig. 14).

Figura 14 – Elevação da frente da camada de ar fria na simulação numérica de um *microburst*, devido a formação de um vórtice secundário próximo a superfície.

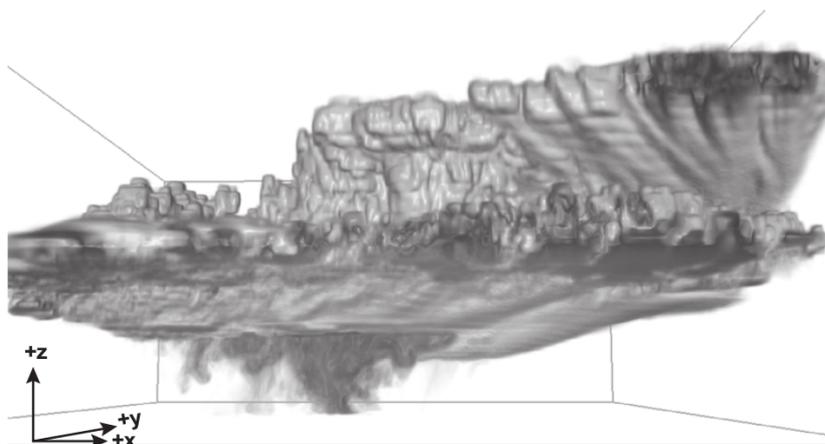


Fonte: Adaptado de Proctor (1988).

Dentre os estudos com o modelo de nuvem completa, Orf, Kantor e Savory (2012) simularam todo ciclo de vida de uma nuvem, ao implementar o modelo de nuvem de Bryan (CM1, Bryan e Fritsch (2002)), em um domínio retangular com dimensões de 92 km x 92 km x 14 km. O domínio possui espaçamento horizontal constante da grade de 20 m, e distância vertical da grade indo de 5 m na superfície, até 95 m no topo, totalizando em 716.800.000 pontos de grade. O modelo CM1 é um modelo numérico não hidrostático, não linear, para estudos de fenômenos da atmosfera com escalas relativamente pequenas, tais como tempestades, downbursts e furacões. Os resultados obtidos por Orf, Kantor e Savory (2012) apresentam características condizentes com tempestades reais (Fig. 15), onde a nuvem *cumulonimbus* gerou vários *microbursts*. Ademais, as ocorrências foram assimétricas, com variações locais, e um desenvolvimento complexo. Por outro lado, a maioria dos modelos mais simples, que utilizam fonte de resfriamento, possuem característica axissimétrica. Diante disso, os autores concluíram que esses modelos não conseguem reproduzir adequadamente a estrutura do escoamento do fenômeno (ORF; KANTOR; SAVORY, 2012).

Dando continuidade a esse trabalho, Orf et al. (2014) realizaram comparações dos dados desse modelo com outros mais simples, aplicando uma análise circunferencial do efluxo do *microburst*, nos instantes das máximas velocidades horizontais registradas. Dentre as suas análises, observaram que essa média circunferencial preservou as regiões das máximas velocidades, havendo uma relação entre as máximas e a velocidade média. Dessa maneira, concluíram que os parâmetros máximos do *microburst* podem ser estimados ao utilizar dados de saída de modelos de nuvem axissimétricos, juntamente com uma alta resolução da malha.

Figura 15 – Ilustração do volume da nuvem cúmulonimbus na simulação numérica. A região escura junto a nuvem que atinge o solo, representa a chuva da tempestade. Esse momento analisado é instantes antes dos ventos atingirem a máxima velocidade próximo da superfície.



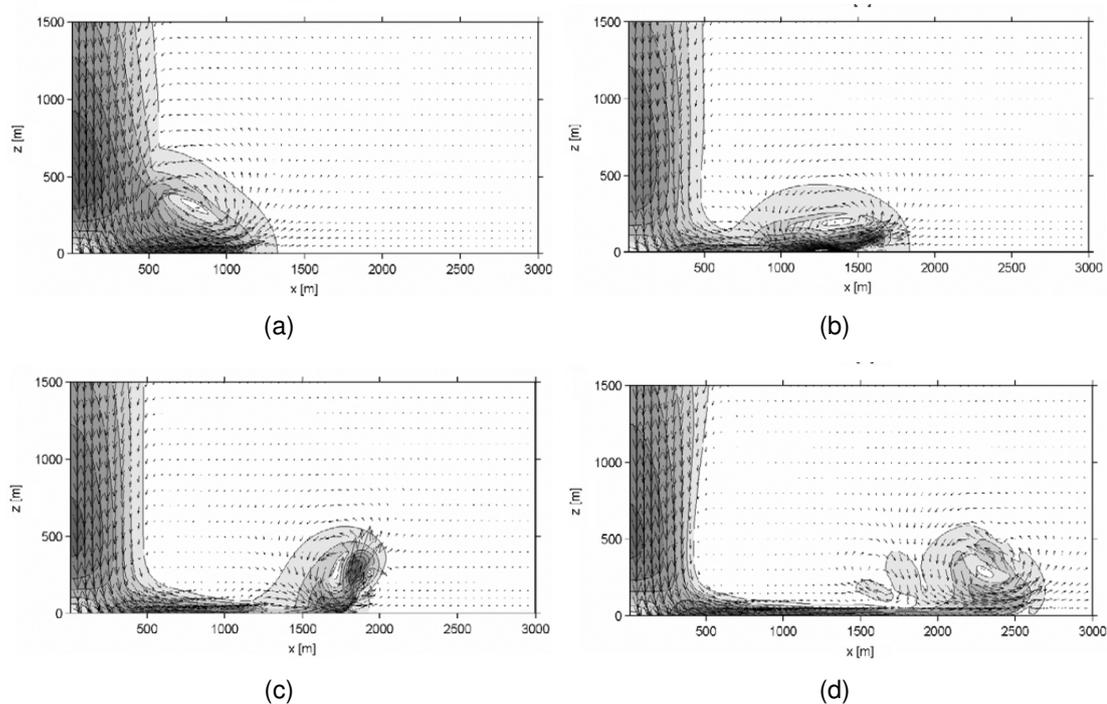
Fonte: Adaptado de Orf, Kantor e Savory (2012).

O custo computacional necessário para realizar a simulação com esse modelo é muito elevado, tornando a sua utilização para análises da engenharia ainda inviável. Portanto, é necessário o desenvolvimento de outros modelos mais simples, tais como o modelo de fonte de resfriamento (ORESKOVIC, 2016).

Anderson, Orf e Straka (1992) com objetivo de aproximar o escoamento de saída de um *microburst* a um modelo de nuvem completa, propôs uma alternativa de análise através de um termo fonte na equação de energia. A justificativa para essa aproximação é que após as análises realizadas em Straka e Anderson (1993), os principais mecanismos de força observados eram termodinâmicos (fusão e evaporação). O termo fonte é dado por uma geometria elíptica, com base na estrutura observada no modelo de nuvem completa, sendo uma função dependente do espaço e do tempo.

Dentre os estudos realizados utilizando esse modelo, Mason, Wood e Fletcher (2009) desenvolveram diversas análises de um *microburst* estático em uma simulação 2D, com modelo de turbulência URANS. Os resultados mostram similaridades com os observados por Proctor (1988). A camada de ar frio descendente forma um anel de vórtice antes de tocar o solo, produzindo jatos de baixo nível logo após tocar a superfície, que são intensificados conforme a camada de ar descende. Após o alastramento radial da camada de ar sobre o solo, o atrito entre essas, forma um vórtice secundário, o qual após se desenvolver, eleva a frente da camada. Essa mudança produz ventos verticais intensos em altitudes maiores, diminuindo a divergência do anel de vórtice, devido a transferência de momento na vertical. O vórtice secundário é dissipado logo após ser jogado para cima, movendo-se sobre o vórtice primário (Fig. 16). No instante da Fig. 16 d, a intensidade dos ventos são aproximadamente 40 % menores que as máximas registradas.

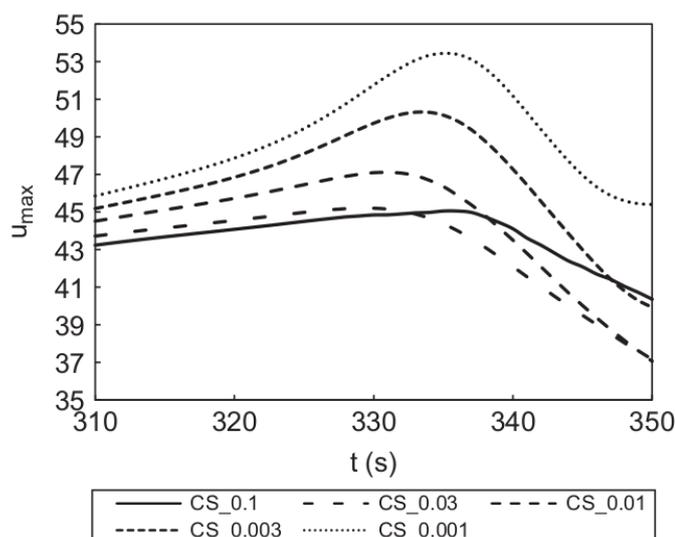
Figura 16 – Evolução temporal do campo de velocidade de um *microburst*, utilizando o modelo de fonte de resfriamento. Os instantes analisados são: (a) $t = 300$ s, (b) $t = 338$ s, (c) $t = 370$ s e (d) $t = 500$ s.



Fonte: Adaptado de Mason, Wood e Fletcher (2009).

Nos estudos desenvolvidos por Vermeire, Orf e Savory (2011a) utilizando o modelo de fonte de resfriamento, realizaram simulações tridimensionais com o modelo de turbulência LES, para diferentes rugosidades do solo. As análises mostram que as velocidades máximas para as diferentes rugosidades, aumentam gradualmente até alcançar um pico, onde como esperado, quanto menor a rugosidade, mais intensas são as velocidades. Essas máximas encontraram-se em um intervalo de tempo entre 310 s e 350 s, após o início da simulação numérica (Fig. 17).

Figura 17 – Velocidades máximas para diferentes rugosidades ao longo do tempo, utilizando o modelo de fonte de resfriamento. Os valores na legenda representam as rugosidades utilizadas no caso.



Fonte: Adaptado de Vermeire, Orf e Savory (2011a).

Para averiguar a influência de diferentes dimensões do termo fonte, assim como outros parâmetros térmicos do modelo, Oreskovic, Orf e Savory (2018) realizaram uma série de análises numéricas utilizando modelo de turbulência LES. Além de diferentes taxas de aspecto da elipse, utilizaram diferentes picos da taxa de resfriamento, $g(t)$, dadas por: $-0,04 \text{ K s}^{-1}$, $-0,06 \text{ K s}^{-1}$, $-0,08 \text{ K s}^{-1}$ e $-0,10 \text{ K s}^{-1}$. Inicialmente Anderson, Orf e Straka (1992) utilizaram o valor de $g(t) = -0,052 \text{ C s}^{-1}$ para a taxa de resfriamento, porém, estudos mais recentes implementaram $g(t) = -0,08 \text{ K s}^{-1}$, para gerar *microbursts* mais intensos (MASON; WOOD; FLETCHER, 2009; ANABOR et al., 2011; VERMEIRE; ORF; SAVORY, 2011a).

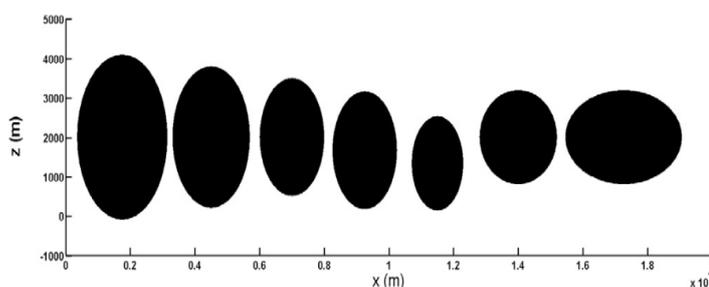
Dentre as análises realizadas por Oreskovic, Orf e Savory (2018), observaram os efeitos gerados pela variação dos diferentes picos da taxa de resfriamento descritos acima, mantendo mesmas dimensões do termo fonte e da altura do centro da elipse. Através dessa análise, constataram que a posição radial e a altura onde acontecem a máxima velocidade radial, não mudavam. Contudo, houve uma relação cúbica entre as velocidades radiais máximas, onde quanto maior a taxa de resfriamento, maior era a velocidade.

Oreskovic, Orf e Savory (2018) observaram ainda as características próximo da superfície, ao variar as dimensões do termo fonte, mantendo a taxa de resfriamento e a altura do centro da elipse constantes. As metades das larguras da elipse nas direções horizontais, foram consideradas iguais ($h_x = h_y$), onde utilizaram em suas análises: $h_x = 1000 \text{ m}$, 1200 m e 1400 m . Para os três casos manteve-se mesma taxa de proporção com a metade da altura da elipse (h_z), dada por $h_z/h_x = 1,5$. Diante dos resultados, as alturas onde as máximas velocidades radiais aconteceram, não apresentaram variação.

Entretanto, as distâncias horizontais onde a máxima velocidade radial ocorreu, tiveram uma relação linear entre os três casos, apresentando a maior distância para o caso $h_x = 1400$ m.

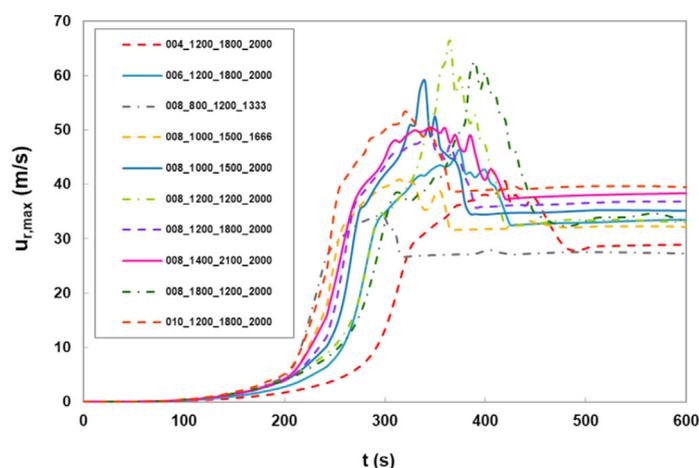
Outra análise relevante investigada por esse trabalho, foi a evolução temporal da máxima velocidade radial para diferentes dimensões (Fig. 18) e taxas de resfriamento do termo fonte. Todos casos apresentaram um aumento gradual da velocidade máxima, até alcançar um pico, e posteriormente começaram perder intensidade. Alguns dos picos exibiram uma forma arredondada, enquanto outros de forma mais aguda (ORESKOVIC; ORF; SAVORY, 2018). A Fig. 19 mostra que ao modificar essas propriedades, houve uma grande diferença dos valores máximos das velocidades, variando de 30 até 65 m s^{-1} , em diferentes instantes de tempo.

Figura 18 – Diferentes geometrias da fonte de resfriamento utilizadas nas simulações numéricas.



Fonte: Adaptado de Oreskovic, Orf e Savory (2018).

Figura 19 – Evolução temporal das velocidades radiais máximas, para diferentes características do termo fonte. A legenda é representada por $g(t)_{h_x}_{h_y}_{h_z}$.



Fonte: Adaptado de Oreskovic, Orf e Savory (2018).

3 METODOLOGIA

3.1 Equações governantes

O modelo utilizado no presente trabalho é um modelo de fonte de resfriamento, não hidrostático, para análises de microburst seco. Essa característica impõe que não haverá presença, nem cálculo de água no caso (MASON, 2009). Assim, as equações de conservação da massa, momento e energia, são descritas por:

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \quad (21a)$$

$$\frac{\partial \bar{u}_j}{\partial t} + \frac{\partial(\bar{u}_i \bar{u}_j)}{\partial x_i} = -\frac{\partial \tilde{p}}{\partial x_j} + \frac{(\rho - \rho_o)}{\rho_o} g_j + \frac{\partial}{\partial x_i} \left[\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] - \frac{\partial \tau_{ij}^r}{\partial x_i} \quad (21b)$$

$$\frac{\partial \bar{\theta}}{\partial t} + \frac{\partial(\bar{\theta} \bar{u}_j)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[\alpha \frac{\partial \bar{\theta}}{\partial x_j} \right] - \frac{\partial \tau_{\theta j}}{\partial x_j} + Q(x, y, z, t) \quad (21c)$$

onde $\tilde{p} = \bar{p} - g\vec{r}$, sendo \bar{p} a pressão cinemática e \vec{r} um vetor posição.

O acoplamento entre as equações de momento e energia é realizada através da aproximação de Boussinesq:

$$\frac{(\rho - \rho_o)}{\rho_o} = -\frac{(\theta - \theta_o)}{\theta_o} \quad (22)$$

Devido aos ventos intensos gerados pelo *microburst*, o fenômeno possui alto valor do número de Rosby. Esse parâmetro mede a importância relativa da aceleração do movimento em relação à força de Coriolis. Portanto, a força inercial de Coriolis é negligenciada nas análises (MASON; WOOD; FLETCHER, 2009). Os efeitos de troca de temperatura da superfície com o ambiente são desconsiderados. Por simplicidade, ρ_o e θ_o são considerados constantes neste trabalho. Enquanto que, em estudos similares, como o de Mason, Wood e Fletcher (2009), utilizaram perfis iniciais para essas variáveis, sendo implementados através da lei do gás ideal e da definição de temperatura potencial.

A partir do termo fonte ($Q(x, y, z, t)$) é imposta a função para reproduzir o efluxo do *microburst*. Esse termo é escrito como:

$$Q(x, y, z, t) = \begin{cases} g(t) \cos^2(\pi R) & \text{se } R \leq 0,5 \\ 0 & \text{se } R > 0,5 \end{cases} \quad (23)$$

Como a geometria formada pelo termo fonte é uma elipse, R é dado por:

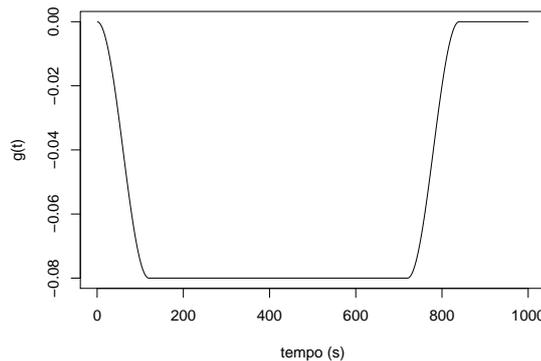
$$R = \sqrt{\left(\frac{x - x_c}{h_x} \right)^2 + \left(\frac{y - y_c}{h_y} \right)^2 + \left(\frac{z - z_c}{h_z} \right)^2} \quad (24)$$

onde x , y e z são as coordenadas ao longo de todo domínio computacional, x_c , y_c e z_c são as coordenadas do centro da elipse, e as variáveis h_x , h_y e h_z são as distâncias do centro da elipse até os seus extremos. A variável $g(t)$ representa o termo forçante temporal, o qual é dado por (ANDERSON; ORF; STRAKA, 1992):

$$g(t) = \begin{cases} -\cos^2 \left[\pi \left(\frac{t-120}{2(\tau)} \right) \right] & \text{se } t < 120 \\ -1 & \text{se } 120 < t < 720 \\ -\cos^2 \left[\pi \left(\frac{t-720}{2(\tau)} \right) \right] & \text{se } 720 < t < 840 \\ 0 & \text{se } t > 840 \end{cases} \quad (25)$$

onde $\tau = 120$ s. A máxima taxa de resfriamento utilizada para esse trabalho é de $g(t) = -0.08 \text{ Ks}^{-1}$, a qual é relacionada com a Eq. 25, ao ser multiplicada pelos termos (Fig. 20).

Figura 20 – Variação do forçante térmico ao longo do tempo, para máxima taxa de resfriamento de $-0,08 \text{ Ks}^{-1}$.

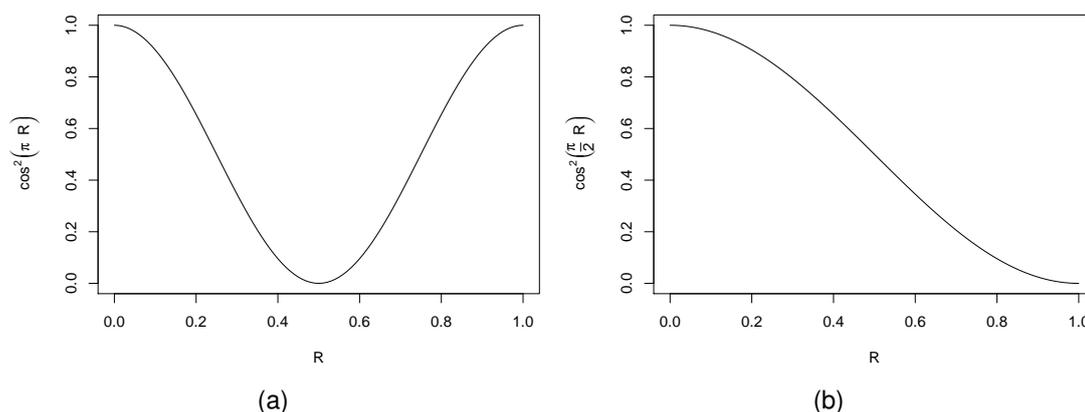


Fonte: Elaborado pelo autor.

Devido à ação do termo fonte, na Eq. 21c, a temperatura começa decair dentro da elipse. A temperatura mínima será em seu centro, aumentando gradualmente ao longo da elipse até alcançar a temperatura ambiente. Oreskovic, Orf e Savory (2018) descrevem que a condição para $Q(x, y, z, t)$, representada por uma elipse, deve possuir limite para valores de R abaixo de 1, e não 0,5, utilizada em estudos anteriores (MASON; WOOD; FLETCHER, 2009; VERMEIRE; ORF; SAVORY, 2011b; ZHANG; HU; SARKAR, 2013). De maneira que, os autores salientam que esse foi apenas um erro no texto, mas que todas equações desses trabalhos foram implementadas corretamente. Entretanto, a Fig. 21a mostra que ao impor $R < 1$ para $\cos^2(\pi R)$, a temperatura não decai de acordo com o proposto pelo modelo. Para tanto, é necessário realizar mais uma alteração na Eq. 23 (Fig. 21b):

$$Q(x, y, z, t) = \begin{cases} g(t) \cos^2\left(\frac{\pi}{2}R\right) & \text{se } R \leq 1 \\ 0 & \text{se } R > 1 \end{cases} \quad (26)$$

Figura 21 – Variação espacial do termo fonte ao longo do raio da elipse, para diferentes expressões da função \cos^2 .



Fonte: Elaborado pelo autor.

3.2 Modelos de subgrade

Os códigos do OpenFOAM são escritos na linguagem de programação C++ e o software utiliza o método dos volumes finitos para discretização das equações. O usuário tem à disposição um vasto bibliotecário base, com opções de pré e pós-processamento, além da possibilidade de desenvolvimento de novos modelos de acordo com a necessidade.

Os modelos de subgrade utilizados neste trabalho foram desenvolvidos pelo *National Renewable Energy Laboratory* (NREL), que disponibiliza um conjunto de ferramentas CFD baseados em OpenFOAM, denominado SOWFA. O SOWFA apresenta diferentes opções de solucionador para análises da camada limite atmosférica, assim como análises da carga de vento sobre parques eólicos (CHURCHFIELD, 2017). Os códigos do SOWFA foram desenvolvidos na versão do OpenFOAM 2.4, dessa maneira, essa é a versão utilizada nas simulações numéricas deste trabalho.

Os modelos utilizados consistem em dois modelos lineares e um modelo não-linear. O modelo não-linear utilizado implementa um termo adicional em conjunto com a hipótese da viscosidade turbulenta de Boussinesq, sendo capaz de reproduzir as características anisotrópicas e de retroespalhamento devido ao cisalhamento de um escoamento. A condição de retroespalhamento impõe que pode acontecer um fluxo de energia local reverso nas trocas de energia, ou seja, transferência de energia das pequenas escalas para as maiores de forma intermitente (PIOMELLI et al., 1991; DOMARADZKI; LIU; BRACHET, 1993). Enquanto que, os modelos lineares possuem somente característica dissipativa, ou

seja, transferência de energia das maiores escalas de turbulência para as menores.

3.2.1 Modelo SmagorinskyABL

Dentre os modelos lineares utilizados está o Smagorinsky, o qual considera equilíbrio local entre a produção e dissipação de energia em escala da subgrade (SMAGORINSKY, 1963). Contudo, o modelo é modificado para implementar dependência da estabilidade atmosférica local, passando a ser denominado como *SmagorinskyABL*. A partir da hipótese da viscosidade turbulenta de Boussinesq, a parte anisotrópica do tensor das tensões residuais desse modelo é relacionada por (POPE, 2001):

$$\tau_{ij}^r = -2\nu_t \bar{S}_{ij} \quad (27a)$$

$$\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j} \right) \quad (27b)$$

onde ν_t é a viscosidade cinemática turbulenta. O cálculo desse termo no modelo *SmagorinskyABL* é de maneira similar ao utilizado pelo OpenFOAM, dado por (OPENCDF, 2020):

$$\nu_t = C_k \Delta \sqrt{k_{sgs}} \quad (28)$$

onde Δ representa a escala de comprimento característico de subgrade e k_{sgs} a energia cinética turbulenta residual. A partir da condição de equilíbrio local imposta pelo modelo, a relação de k_{sgs} é dada por:

$$k_{sgs} = 2 \frac{C_k}{C_e} \Delta^2 \left| \left(\frac{1}{2} (\nabla U + (\nabla U)^T) - \frac{1}{3} tr(\nabla U) I \right) \right|^2 \quad (29a)$$

$$k_{sgs} = 2 \frac{C_k}{C_e} \Delta^2 |dev[symm(\nabla U)]|^2 \quad (29b)$$

onde, C_k e C_e são constantes do modelo SGS. Os valores utilizados neste trabalho são $C_k = 0.0673$ e $C_e = 0.93$ (REN; LAIMA; LI, 2020).

A parametrização do tensor do fluxo turbulento de calor de subgrade é proporcional ao gradiente da temperatura potencial filtrada (MOENG, 1984):

$$\tau_{\theta j} = -\alpha_t \frac{\partial \bar{\theta}}{\partial x_j} \quad (30)$$

onde α_t é a difusividade térmica turbulenta, dada por $\alpha_t = \nu_t / Pr_t$. O termo Pr_t representa o número de Prandtl turbulento. As relações impostas para as variáveis Δ e Pr_t são descritas com mais detalhes na sessão 3.3.

3.2.2 Modelo oneEqEddyABL

O segundo modelo SGS utilizado é baseado no *oneEqEddy* do OpenFOAM, todavia, são feitas algumas pequenas modificações. O termo de produção de flutuabilidade é adaptado para escoamento atmosférico, e o nome do modelo passa a ser representado por *oneEqEddyABL*. A parametrização dos tensores τ_{ij}^r e $\tau_{\theta j}$ desse modelo são iguais aos do modelo *SmagorinskyABL* (Eqs. 27a e 30). Contudo, o cálculo da viscosidade cinemática turbulenta difere desse modelo, o qual é dado por:

$$\nu_t = C_k l \sqrt{k_{sgs}} \quad (31)$$

onde l representa o comprimento característico do número de Prandtl turbulento, a qual é descrita com mais detalhes na sessão 3.3. O valor de C_k utilizado é o mesmo do *SmagorinskyABL*.

Diferente do modelo *SmagorinskyABL*, esse modelo apresenta uma equação prognóstica para a energia cinética turbulenta, dada por:

$$\begin{aligned} \frac{\partial k_{sgs}}{\partial t} + \frac{(\bar{u}_j k_{sgs})}{\partial x_j} - \frac{\partial}{\partial x_j} \left[2(\nu + \nu_t) \frac{\partial k_{sgs}}{\partial x_j} \right] = & 2\nu_t \left[\frac{1}{2} (\nabla u + \nabla u^T) \right]^2 \\ & + \frac{g}{\bar{\theta}_o} \frac{\nu_t}{Pr_t} \frac{\partial \bar{\theta}}{\partial x_j} - \frac{C_\epsilon k_{sgs}^{3/2}}{\Delta} \end{aligned} \quad (32)$$

onde os termos da esquerda da igualdade representam, respectivamente, a taxa de variação local, termo convectivo e o termo difusivo; os termos da direita da igualdade, a primeira expressão representa o termo de produção devido ao cisalhamento, o segundo devido a flutuabilidade e o último denota a taxa de dissipação.

Moeng (1984) ao realizar estudos da camada limite atmosférica com o modelo LES, utilizou inicialmente a relação de $C_\epsilon = 0,19 + (0,51l/\Delta)$. Contudo, posteriormente declarou o valor sendo $C_\epsilon = 0,93$ (MOENG; WYNGAARD, 1988), onde esse valor tinha sido definido anteriormente também por Lilly (1966). Diante disso, SOWFA reformula essa variável para que seja revisada, mantendo similaridade com a forma original de Moeng (1984). De maneira que, ela permite ainda a possibilidade de escolha de um valor arbitrário, quando os comprimentos característicos são iguais:

$$C_\epsilon = \frac{C'_\epsilon}{0,93} \left(0,19 + \left(0,74 \frac{l}{\Delta} \right) \right) \quad (33)$$

utiliza-se para esse trabalho $C'_\epsilon = 0,93$. Para evitar que os valores da energia cinética turbulenta tornem-se indevidamente grandes na superfície, mais uma condição é determinada para essa variável. O modelo *oneEqEddyABL* estabelece para o nível mais baixo do domínio o valor de $C_\epsilon = 3,9$ (DEARDORFF, 1980).

3.2.3 Modelo KosovicOneEqNBA

O terceiro modelo empregado é denominado no SOWFA como *KosovicOneEqNBA*, onde como dito anteriormente, esse modelo não-linear é capaz de reproduzir as características anisotrópicas e de retroespalhamento devido ao cisalhamento de um escoamento (KOSOVIĆ, 1997; KOSOVIĆ; CURRY, 2000). A parametrização do tensor $\tau_{\theta j}$ é igual aos modelos lineares (Eq. 30), contudo, o tensor anisotrópico das tensões residuais desse modelo é dado por:

$$\tau_{ij}^r = -2\nu_t \bar{S}_{ij} + M_{ij} \quad (34)$$

O cálculo da viscosidade cinemática turbulenta desse modelo SGS é relacionado por:

$$\nu_t = C_e \Delta \sqrt{k_{sgs}} \quad (35)$$

A equação de transporte da energia cinética turbulenta desse modelo é similar ao modelo *oneEqEddyABL*, onde as diferenças estão no termo de produção devido ao cisalhamento, e no termo de taxa de dissipação:

$$\frac{\partial k_{sgs}}{\partial t} + \frac{(\bar{u}_j k_{sgs})}{\partial x_j} - \frac{\partial}{\partial x_j} \left[2(\nu + \nu_t) \frac{\partial k_{sgs}}{\partial x_j} \right] = -\tau_{ij} \frac{\partial \bar{u}_i}{\partial x_j} + \frac{g}{\bar{\theta}_o} \frac{\nu_t}{Pr_t} \frac{\partial \bar{\theta}}{\partial x_j} - \frac{C_\epsilon k_{sgs}^{3/2}}{l_\epsilon} \quad (36)$$

Por sua vez, o tensor de tensões da SGS modificado pelo modelo, que inclui a não linearidade, é dado por:

$$\tau_{ij} = \frac{2}{3} k_{sgs} \delta_{ij} - \nu_t \left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j} \right) + M_{ij} \quad (37)$$

onde,

$$M_{ij} = -C_e \Delta^2 C_s^{\frac{2}{3}} \left(\frac{27}{8\pi} \right)^{\frac{1}{3}} \times \left[C_1 \left(S_{ik} S_{kj} - \frac{1}{3} S_{mn} S_{nm} \delta_{ij} \right) + C_2 (S_{ik} \Omega_{kj} - \Omega_{ik} S_{kj}) \right] \quad (38)$$

A parte não linear das tensões da SGS é representada por M_{ij} , e os termos S_{ij} e Ω_{ij} são os tensores de taxa de deformação e taxa de rotação, respectivamente. A variável C_e é dada em função do parâmetro de Smagorinsky (C_s), cujas relações são:

$$C_e = \left(\frac{8\pi}{27} \right)^{1/3} C_s^{\frac{4}{3}} \quad (39a)$$

$$C_s = \left[\frac{8(1 + C_b)}{27\pi^2} \right]^{\frac{1}{2}} \quad (39b)$$

a variável C_b representa o parâmetro de retroespalhamento, dado por $C_b = 0,36$. KOSOVIĆ (1997) obteve esse valor através de experimentos, o qual também está de acordo com resultados de DNS e predição teórica.

Para o termo não linear fornecer valores adequados para a transferência de energia, e capturar os efeitos da tensão normal em fluxos cisalhantes e homogêneos, determina-se os parâmetros C_1 e C_2 (KOSOVIĆ; CURRY, 2000). Para reproduzir os efeitos da tensão normal, C_2 é determinado igual a C_1 , pois não é possível obtê-lo por relações da energia cinética turbulenta, já que os tensores de taxa de rotação não contribuem para o balanço da equação (KOSOVIĆ, 1997). Esses parâmetros são dados por:

$$C_1 = -C_2 = \frac{960^{1/2} C_b}{7(1 + C_b) S(k_c)} \quad (40)$$

o termo $S(k_c)$ representa a função de assimetria derivada da velocidade resolvida, cujo valor obtido experimentalmente é dado por $S(k_c) = 0,5$ (KOSOVIĆ; CURRY, 2000).

A constante no termo da taxa de dissipação, C_ϵ , é assumida $C_\epsilon = 0,93$, e a escala de comprimento de dissipação, l_ϵ , é determinada pela seguinte relação:

$$l_\epsilon = \left[\frac{1}{\Delta^2} + \frac{1}{l_n^2} + \frac{1}{l_s^2} \right]^{-\frac{1}{2}} \quad (41)$$

onde, l_n e l_s representam os comprimentos de escala de flutuação e de cisalhamento da SGS, respectivamente, os quais são dados por (KOSOVIĆ; CURRY, 2000):

$$l_n = 0,76 \frac{\sqrt{k_{sgs}}}{\sqrt{\frac{g}{T_o} \left(\frac{\partial \theta}{\partial z} \right)}} \quad (42a)$$

$$l_s = 2,76 \frac{\sqrt{k_{sgs}}}{\sqrt{\left(\frac{\partial u}{\partial z} \right)^2 + \left(\frac{\partial v}{\partial z} \right)^2}} \quad (42b)$$

3.3 Comprimento característico de Pr_t e de subgrade

A relação do número de Prandtl turbulento para os modelos SGS *SmagorinskyABL* e *oneEqEddyABL*, é representada por (MOENG, 1984):

$$Pr_t = \frac{1}{\left(1 + 2 \frac{l}{\Delta}\right)} \quad (43)$$

Moeng (1984) relaciona l através da frequência de Brunt-Väisälä. Esse parâmetro é definido como a frequência em que uma parcela de ar se movimentará em uma atmosfera estavelmente estratificada, sobre a aplicação de uma força infinitesimal, representando uma medida de estabilidade (DURRAN; KLEMP, 1982). Esse comprimento característico é expresso por:

$$l = \begin{cases} \min \left(0,76\sqrt{k_{sgs}}\sqrt{\frac{1}{s}}, \Delta \right) & \text{se } s > 0 \\ \Delta & \text{se } s \leq 0 \end{cases} \quad (44a)$$

$$s = \frac{|g_i|}{\theta_0} \frac{\partial \bar{\theta}}{\partial z} \quad (44b)$$

A estabilidade local atmosférica é classificada de acordo com a Tab. 1, onde Θ_v representa a temperatura potencial virtual, que é definida como a temperatura que o ar seco deve ter para se igualar à densidade do ar úmido na mesma pressão (STULL, 1988). A Tab. 1, mostra que para estabilidade local estável, o comprimento característico de Pr_t é dado para $s > 0$, e para as estabilidades locais instável e neutra, definido para $s \leq 0$.

Tabela 1 – Classificações da estabilidade local atmosférica.

	$\partial\Theta_v/\partial z$	Estabilidade estática local
Superadiabático	< 0	Instável
Adiabático	= 0	Neutra
Subadiabático	> 0	Estável

Fonte: Adaptado de (STULL, 1991).

O cálculo do número de Prandtl turbulento do modelo *KosovicOneEqNBA* é de maneira similar aos modelos lineares, contudo, ele é dado em função da escala de comprimento de dissipação:

$$Pr_t = \frac{1}{\left(1 + 2\frac{l_\epsilon}{\Delta}\right)} \quad (45)$$

Para o cálculo da escala de comprimento característica de subgrade, utiliza-se a mesma função para todos modelos SGS. A função empregada é a de amortecimento de Van Driest (DRIEST, 1956), disponível no software OpenFOAM, que tem como finalidade diminuir o valor da viscosidade da subgrade próximo à parede (FERZIGER; PERIĆ, 2002). Essa função é aplicada até a região de $y^+ < 500$ (OPENCDF, 2020). A sua relação é dada por:

$$\Delta = \min \left(\frac{\kappa y}{C_\Delta} \left(1 - e^{-\frac{y^+}{A^+}}\right), \Delta_g \right) \quad (46)$$

onde $\kappa = 0,41$ é a constante de von Kármán, os parâmetros C_Δ e A^+ por padrão do software CFD ($C_\Delta = 0,158$ e $A^+ = 26$), y representa a altura em relação a superfície, Δ_g é uma função dada pela raiz cúbica do volume ($\Delta = c(V_c)^{\frac{1}{3}}$), sendo c uma constante, cujo valor é $c = 1$, e V_c representa o volume da grade.

O comprimento característico de subgrade utilizado por Moeng (1984) é igual a $\Delta =$

$(\Delta_x \Delta_y \Delta_z)^{1/3}$. Contudo, utiliza-se o modelo Van Driest neste trabalho, devido ao modelo da raiz cúbica do volume ser válido para taxa de aspecto da malha igual a 1 (OPENCDF, 2020). Esse parâmetro mede a relação entre o comprimento mais longo e mais curto da grade. De maneira que, após verificação da malha neste trabalho, a malha possui valor máximo igual a 10.

3.4 Modelo de parede

O modelo de parede utilizado no presente trabalho, advém do conjunto de ferramentas do SOWFA. A partir de um modelo as tensões viscosas e da SGS na superfície são parametrizadas. Inicialmente, assume-se que o centro das primeiras células adjacentes à superfície, encontram-se na camada superficial da camada limite atmosférica, então, as tensões na superfície são representadas por (CHURCHFIELD, 2017):

$$\tau_{ij}^D = \begin{bmatrix} 0 & 0 & \tau_{13}^{tot} \\ 0 & 0 & \tau_{23}^{tot} \\ \tau_{13}^{tot} & \tau_{23}^{tot} & 0 \end{bmatrix}.$$

Para parametrizar essas tensões é utilizado o modelo Schumann-Grötzbach (SCHUMANN, 1975; GRÖTZBACH, 1987). O modelo é utilizado em simulações LES com alto valor de Reynolds, além de ser um dos mais utilizados para análises em terrenos planos (ABKAR; PORTÉ-AGEL, 2012). Esse modelo é representado pela seguinte relação:

$$\tau_{13}^{tot} = -u_*^2 \frac{(\bar{u}_{1/2} - \langle \bar{u}_{1/2} \rangle)}{(\langle \bar{u}_{1/2} \rangle^2 + \langle \bar{v}_{1/2} \rangle^2)^{\frac{1}{2}}} \quad (47)$$

$$\tau_{23}^{tot} = -u_*^2 \frac{(\bar{v}_{1/2} - \langle \bar{v}_{1/2} \rangle)}{(\langle \bar{u}_{1/2} \rangle^2 + \langle \bar{v}_{1/2} \rangle^2)^{\frac{1}{2}}} \quad (48)$$

onde os subscritos 1/2 são os valores das variáveis referentes ao centro das primeiras células adjacentes à superfície. Os parâmetros dados por $\langle \rangle$, denotam a média dessas variáveis em relação ao plano paralelo à superfície, na altura do centro das primeiras células. A velocidade de fricção é definida por $u_* = (\langle \tau_{13}^{tot} \rangle^2 + \langle \tau_{23}^{tot} \rangle^2)^{1/2}$. Para adicionar os efeitos da rugosidade ao cálculo de u_* , utiliza-se a lei exponencial para parede rugosa, onde ao rearranjar os termos, passa a ser dada por (CHURCHFIELD, 2017):

$$\frac{(\langle \bar{u}_{1/2} \rangle^2 + \langle \bar{v}_{1/2} \rangle^2)^{\frac{1}{2}}}{u_*} = \frac{1}{\kappa} \ln \left(\frac{z}{z_o} + f(L) \right) \quad (49)$$

o termo $f(L)$ é um parâmetro de estabilidade atmosférico, onde L é o comprimento de Obuhkov (LALAS; RATTO, 1996). A rugosidade implementada para todas simulações numéricas do trabalho é dada por $z_o = 0,01$ m. Esse valor equivale a planícies de grama razoavelmente niveladas (STULL, 1988).

3.5 Implementação da fonte de resfriamento

A implementação do forçante térmico no OpenFoam para a geração do microburst, é auferida por intermédio da funcionalidade fvOptions. Essa funcionalidade possui um conjunto de opções com diferentes aplicabilidades de termo fonte, assim como a possibilidade de criação de novas funções, sem a necessidade de alterar os códigos do *solver* e compilar novamente. A opção para desenvolver um novo código de um campo escalar no fvOptions é por intermédio da função *scalarCodedSource*. Essa funcionalidade não precisa ser compilada, após desenvolver o código ela gera automaticamente no momento que tem início as discretizações das equações. Abaixo está demonstrado o código responsável pela implementação do termo fonte na equação de energia:

```

1  sourceTemperature
2  {
3      type            scalarCodedSource;
4      active          true;
5      selectionMode   all;
6      rampCoeffs      {};
7      scalarCodedSourceCoeffs
8      {
9          fieldNames      (T);
10
11         // Name of generated source
12         redirectType     ramp;
13
14         // Include additional header files
15         codeInclude
16         #{
17         #};
18
19         // Apply corrections after the equation has been solved
20         codeCorrect
21         #{
22             Pout<< "**codeCorrect**" << endl;
23         #};
24
25         // Include sources
26         codeAddSup
27         #{
28             const Time& time = mesh().time();
29             const volVectorField& ctrs = mesh_.C();
30             scalarField& QSource = eqn.source();
31             const scalarField& V = mesh_.V();
32             scalar    Xo = 0.0;

```

```

33     scalar   Yo = 0.0;
34     scalar   Zo = 2000.0;
35     scalar   hx = 1200.0;
36     scalar   hy = 1200.0;
37     scalar   hz = 1800.0;
38     scalar   gt = 0.08;
39
40     forAll(ctr, i)
41     {
42         scalar   sqrX = sqrt((ctr[i].component(vector::X) - Xo)/hx);
43         scalar   sqrY = sqrt((ctr[i].component(vector::Y) - Yo)/hy);
44         scalar   sqrZ = sqrt((ctr[i].component(vector::Z) - Zo)/hz);
45         scalar   R = sqrt(sqrX + sqrZ + sqrY);
46
47         if ((time.value() >= 0) && (time.value() < 120))
48         {
49             gt = 0.08*sqrt(cos(M_PI*((time.value() - 120)/240)));
50         }
51         else if ((time.value() >= 120) && (time.value() <= 720))
52         {
53             gt = 0.08;
54         }
55         else if ((time.value() > 720) && (time.value() <= 840))
56         {
57             gt = 0.08*sqrt(cos(M_PI*((time.value() - 720)/240)));
58         }
59         else if (time.value() > 840)
60         {
61             gt = 0.0;
62         }
63         if (R <= 1)
64         {
65             QSource[i] = gt*sqrt(cos((M_PI*R)/2))*V[i];
66         }
67         else
68         {
69             QSource[i] = 0.0;
70         }
71     }
72     #};
73
74     // Constrain values before the equation is solved
75     codeSetValue
76     #{
77         Pout<< "***codeSetValue**" << endl;
78     #};
79

```

```

80         // Dummy entry. Make dependent on above to trigger recompilation
81         code
82         #{
83             $codeInclude
84             $codeCorrect
85             $codeAddSup
86             $codeSetValue
87         #};
88     }
89     sourceTimeCoeffs
90     {
91         $scalarCodedSourceCoeffs ;
92     }
93 }

```

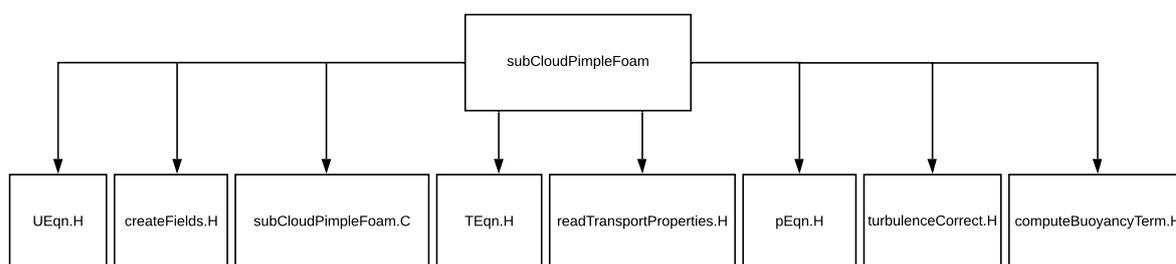
O desenvolvimento do código foi realizado em cima de um exemplo presente no arquivo de cabeçalho, *CodedSource.H*. A opção *scalarCodedSource* faz parte dessa *class*. As principais modificações realizadas para implementar o modelo de fonte de resfriamento foram em *codeAddSup*. As outras opções não foram modificadas, onde comentários com suas descrições foram colocados em cima dessas, para compreensão das suas funcionalidades (OPENCFD, 2020).

O código do modelo tem início com a declaração da variável para tomar o tempo da simulação numérica (linha 28). Após, declara-se a variável que obtém as coordenadas dos centros das células do domínio computacional (linha 29), seguida das variáveis para criar o termo fonte (linha 30), e para obter os volumes das células (linha 31). Depois desses passos, são declaradas as variáveis necessárias para dimensionar a elipse e a taxa de resfriamento (linha 32 até 38). Posteriormente, um laço condicional percorre todas células do domínio computacional (linha 40), seguido das expressões para obtenção da variável R (linha 42 até 45). Diante da taxa de resfriamento ser dependente do tempo (Eq. 25), aplicam-se laços condicionais (linha 47 até 62), onde subsequentemente a variável é utilizada nos laços condicionais para resolução da Eq. 26 (linha 63 até 70).

3.6 Solver

O *solver* utilizado tem como base o *buoyantBoussinesqPimpleFoam* do OpenFOAM, porém, algumas modificações são realizadas. De posse dos códigos do *solver ABLSolver* do SOWFA, algumas condições foram utilizadas em conjunto com o *solver* do OpenFOAM. *ABLSolver* é um *solver* transiente, para escoamento flutuante e turbulento de fluidos incompressíveis, com modificações para análise da camada limite atmosférica. Devido a essas alterações, criou-se um novo *solver* denominado *subCloudPimpleFOAM*. Os arquivos de cabeçalho (extensão .H), e o arquivo fonte (extensão .C) desse *solver*, são exibidos abaixo:

Figura 22 – Representação dos arquivos de cabeçalho e arquivo fonte do *solver subCloudPimpleFOAM*.



Fonte: Elaborado pelo autor.

Dentre as modificações realizadas no novo *solver*, está no arquivo de cabeçalho *createFields.H*, onde foram declaradas as variáveis de parametrização das tensões e do fluxo de calor na superfície.

```

1   volSymmTensorField Rwall
2   (
3     IOobject
4     (
5       "Rwall",
6       runTime.timeName(),
7       mesh,
8       IOobject::MUST_READ,
9       IOobject::AUTO_WRITE
10    ),
11    mesh
12  );
13
14  volVectorField qwall
15  (
16    IOobject
17    (
18      "qwall",
19      runTime.timeName(),
20      mesh,
21      IOobject::MUST_READ,
22      IOobject::AUTO_WRITE
23    ),
24    mesh
25  );
  
```

A parametrização do modelo de parede *SchumannGrotzbach* é auferida por intermédio da variável *Rwall*. O cálculo dessa parametrização é implementada em *UEqn.H*:

```

1  #include "computeBuoyancyTerm.H"
2
3  fvVectorMatrix UEqn
4  (
5      fvm::ddt(U)
6      + fvm::div(phi, U)
7      + turbulence->divDevReff(U)
8      + fvc::div(Rwall)
9      ==
10     fvOptions(U)
11 );
12 UEqn.relax();
13
14 fvOptions.constrain(UEqn);
15
16 if (pimple.momentumPredictor())
17 {
18     solve
19     (
20         UEqn
21         ==
22         fvc::reconstruct
23         (
24             (
25                 - fvc::snGrad(p_rgh)          gradient
26                 + buoyancyTerm
27             ) * mesh.magSf()
28         )
29     );
30
31     fvOptions.correct(U);
32 }

```

Outra modificação realizada em *UEqn.H*, está na implementação de um arquivo de cabeçalho para o termo de flutuabilidade, empregado através de *computeBuoyancyTerm.H*. Esse termo foi colocado de maneira similar ao *ABLSolver*, já que esse apresenta três opções de abordagem da perturbação da pressão. Contudo, a condição imposta para esse trabalho, resulta na relação dada na Eq. 21b, sendo dada por:

```

1 surfaceScalarField buoyancyTerm = -ghf*fvc::snGrad(rhok);
2
3 buoyancyTerm = ((g & mesh.Sf())/mesh.magSf()) * fvc::interpolate(rhok - 1.0);

```

Para realizar o cálculo de *rhok*, presente em *TEqn.H*, modifica-se essa variável do *solver* original, para deixar o caso em função da Eq. 22:

```

1 {
2   volScalarField kappaEff("kappaEff", turbulence->nu()/Pr + kappat);
3
4   fvScalarMatrix TEqn
5   (
6     fvm::ddt(T)
7     + fvm::div(phi, T)
8     - fvm::laplacian(kappaEff, T)
9     - fvc::div(qwall)
10    ==
11    fvOptions(T)
12   );
13
14   TEqn.relax();
15   fvOptions.constrain(TEqn);
16   TEqn.solve();
17   fvOptions.correct(T);
18   rhok = 1.0 - ((T - TRef)/TRef);
19 }

```

A correção da difusividade térmica turbulenta em *TEqn.H* é removida do *solver* original do OpenFOAM. Tanto essa correção, como outras pertinentes ao caso, são auferidas pelo arquivo de cabeçalho *turbulenceCorrect.H*:

```

1 {
2   kappat = turbulence->nut()/Prt;
3   turbulence->correct();
4   kappat.correctBoundaryConditions();
5   qwall.correctBoundaryConditions();
6   Rwall.correctBoundaryConditions();
7 }

```

Essas correções são impostas no arquivo fonte de maneira similar ao *ABLSolver*, onde tanto o arquivo *turbulenceCorrect.H*, como o *TEqn.H*, são adicionados após o código PIMPLE:

```

1   // — Pressure-velocity PIMPLE corrector loop
2   while (pimple.loop())
3   {
4     #include "UEqn.H"
5     #include "turbulenceCorrect.H"
6     #include "TEqn.H"
7
8     // — Pressure corrector loop

```

```
9      while (pimple.correct())
10     {
11         #include "pEqn.H"
12         #include "turbulenceCorrect.H"
13         #include "TEqn.H"
14     }
15 }
```

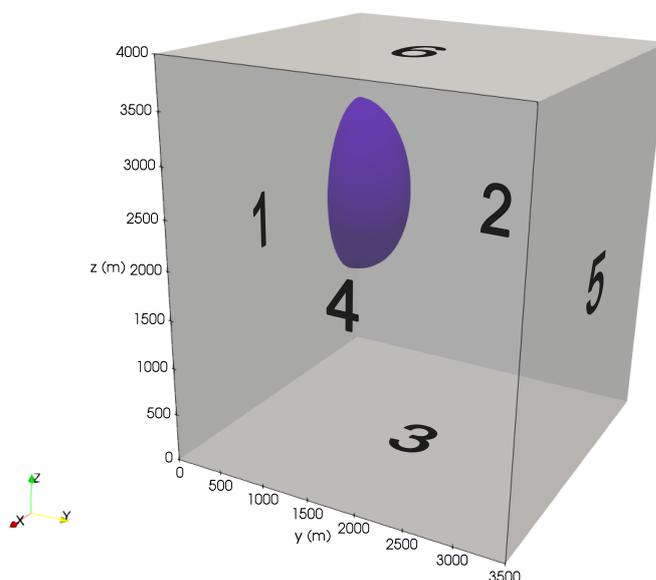
O acoplamento entre pressão-velocidade é dado por *pEqn.H*. Utilizou-se o mesmo código do *ABLSolver*, que é similar ao do *buoyantBoussinesqPimpleFoam*, porém, possui algumas pequenas modificações. Dentre as manipulações, está na adequação do termo de flutuabilidade para o caso. A outra alteração é um balanço de fluxo de massa global, a qual aplica uma correção indiretamente na condição de contorno de *p_rgh*, definindo um gradiente de pressão quando utilizada a condição *fixedFluxPressure*. De maneira que, se for implementado a condição *zeroGradient*, a correção é realizada diretamente.

O método de acoplamento é realizado pelo algoritmo *PIMPLE*, o qual é uma combinação dos métodos semi-implícito para equações ligadas à pressão (tradução livre do inglês “*semi-implicit method for pressure-linked equations*” - SIMPLE) e o método de operador de divisão de pressão implícita (tradução livre do inglês “*pressure-implicit split-operator*” - PISO). O método *PIMPLE* é similar ao algoritmo *PISO*, tendo como diferença a possibilidade de habilitar a resolução de todo sistema mais de uma vez dentro de um passo de tempo, onde ao configurar esse número para 1, o algoritmo replica o método *PISO*.

3.7 Condições iniciais e de contorno

As simulações numéricas desenvolvidas neste trabalho analisam um microburst estático, fazendo o uso de condições de simetria. Dessa maneira, visando diminuir o tempo computacional, configura-se os casos para representarem 1/4 do fenômeno. O domínio computacional possui 4 Km de altura, com uma base quadrada de 3,5 Km de aresta, de maneira que as características tridimensionais da turbulência sejam mantidas, ao analisar um caso 3D (Fig. 23). A divisão da grade possui um refinamento de 1 m de altura nas primeiras células adjacentes à superfície 3 (Fig. 23), com um aumento gradual em seu espaçamento até 70 m no topo do domínio computacional. O espaçamento horizontal é mantido constante, com distanciamento de 10 m em ambas direções dos eixos x e y, totalizando em 29.767.500 de células. É importante destacar que, o espaçamento horizontal é baseado em outros trabalhos que também analisaram as características de propagação do microburst utilizando o modelo de fonte de resfriamento (VERMEIRE; ORF; SAVORY, 2011a, 2011b; ORESKOVIC; ORF; SAVORY, 2018). Contudo, o espaçamento vertical utilizado difere desses, os quais refinaram a grade de 1 m até 50 m no topo do domínio computacional.

Figura 23 – Domínio computacional utilizado no trabalho, com ilustração da fonte de resfriamento em azul.



Fonte: Elaborado pelo autor.

Como condições de contorno, determina-se que as faces 4, 5 e 6 são fronteiras de saída (Fig. 23), as faces adjacentes ao termo fonte, 1 e 2, impõem-se a condição de plano de simetria, restando a face 3, que representa o solo com a condição de parede.

A condição de parede da velocidade utilizada é a *velocityABLWallFunction*. Essa condição advém do SOWFA, sendo utilizada para determinar a velocidade na superfície, em um escoamento incompressível da camada limite atmosférica com rugosidade. Ela tem como objetivo definir um gradiente vertical adequado para utilizar no modelo de turbulência (CHURCHFIELD, 2017). Para as faces de saída são impostas a condição, *pressureInletOutletVelocity*, que atribui a condição de gradiente zero, e considera-se o campo interno da velocidade como nulo.

A temperatura potencial é assumida em todo domínio computacional como 300 K, com a condição de zero gradiente para as fronteiras. A condição de pressão imposta na parte inferior do domínio é a *fixedFluxPressure*. Essa condição define que o gradiente de pressão seja tal que o escoamento próximo à superfície seja especificado pela condição de contorno da velocidade. Para as faces de saída, utiliza-se a condição de pressão, *totalPressure*, que define uma pressão estática para a pressão total. As demais condições de contorno impostas para as outras variáveis encontram-se no Apêndice A.

A altura do centro do termo fonte é de 2000 m, e os valores das dimensões da elipse são $h_x = h_y = 1200$ m e $h_z = 1800$. Esses valores foram utilizados por Anderson, Orf e Straka (1992) quando desenvolveram o modelo, assim como por outros trabalhos subsequentes (ORF; ANDERSON, 1999; VERMEIRE; ORF; SAVORY, 2011a, 2011b). Knupp (1989) em suas análises utilizando modelo de nuvem completa, mostrou que tanto em *mi-*

crobursts secos, como nos úmidos, a altura que os efeitos das correntes descendentes começam atuar é aproximadamente na altura de 2 Km.

De maneira a manter convergência nos casos, o passo de tempo é ajustável, o qual é baseado no número de Courant, que representa o deslocamento de uma partícula ao longo dos volumes de controle, garantindo precisão temporal e estabilidade numérica. O valor utilizado nas simulações numéricas é de $Co = 1$, que estabelece que a partícula percorra de uma célula para outra a cada passo de tempo. O número de Courant no OpenFOAM é dado por:

$$Co = \Delta t \tau \quad (50)$$

onde Δt é o passo de tempo, e τ é a escala de tempo característica, sendo dada por:

$$\tau = \frac{1}{2V} \sum_{faces} |\phi_i| \quad (51)$$

a variável V representa o volume da célula, ϕ é o fluxo volumétrico na face, e o símbolo somatório denomina a soma de todas faces da célula.

Os métodos de discretizações de volumes finitos utilizados são todos de segunda ordem (Tab. 2), com exceção do termo divergente da temperatura potencial, onde utiliza-se *Gauss limitedLinear 1*. Essa função ainda realiza a discretização pelo modelo de segunda ordem, *Gauss linear*, mas nas regiões com rápida mudança do gradiente, a discretização é auferida pelo modelo de primeira ordem *upwind*.

Tabela 2 – Esquemas numéricos utilizados no presente trabalho.

Sub-dicionário	Termo matemático	Esquema numérico
timeScheme	derivativos do tempo: $\partial/\partial t, \partial^2/\partial^2 t$	backward
gradSchemes	gradiente: ∇	Gauss linear
divSchemes	divergente: $\nabla \bullet$	Gauss linear
laplacianSchemes	laplaciano: ∇^2	Gauss linear corrected
interpolationSchemes	Interpolações ponto a ponto de valores	linear
snGradSchemes	Componente do gradiente normal para uma face celular	corrected

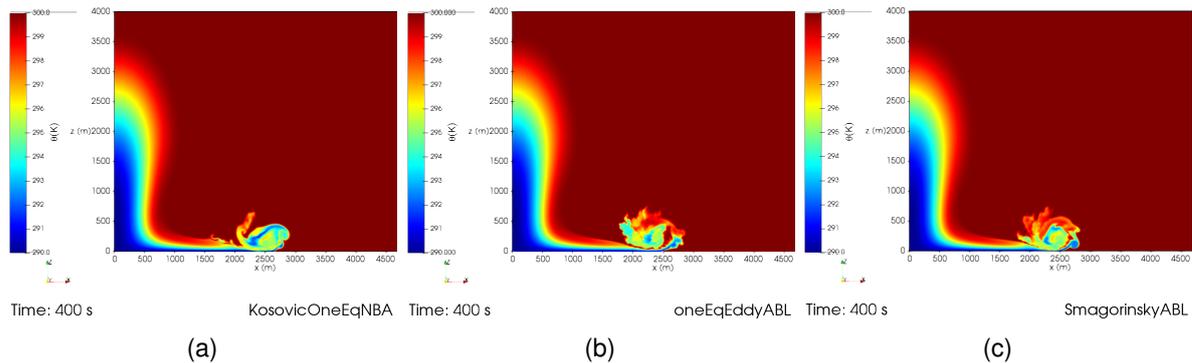
Fonte: Elaborado pelo autor.

4 RESULTADOS E DISCUSSÕES

Como já mencionado, o objetivo deste trabalho é implementar o modelo de fonte de resfriamento e estudar a capacidade de reprodução do efluxo de um *microburst*, e analisar a influência de diferentes parametrizações de turbulência na propagação da frente de onda. Para o caso de um *microburst* isolado e estático, as principais características a serem observadas no trabalho são quanto a estrutura e aumento da difusividade nas frentes de ondas dos diferentes modelos, e análise das velocidades máximas dos ventos na propagação dos *microbursts*.

Após o resfriamento da elipse, forma-se uma coluna de ar fria descendente, devido a diferença de massa específica em relação ao meio circundante. A camada de ar fria ao escoar sobre o solo, concentra as menores temperaturas nos níveis mais baixos. Conforme o anel de vórtice escoar sobre a superfície, sua rotação no sentido anti-horário faz com que as baixas temperaturas também alcancem níveis mais altos (Fig. 24).

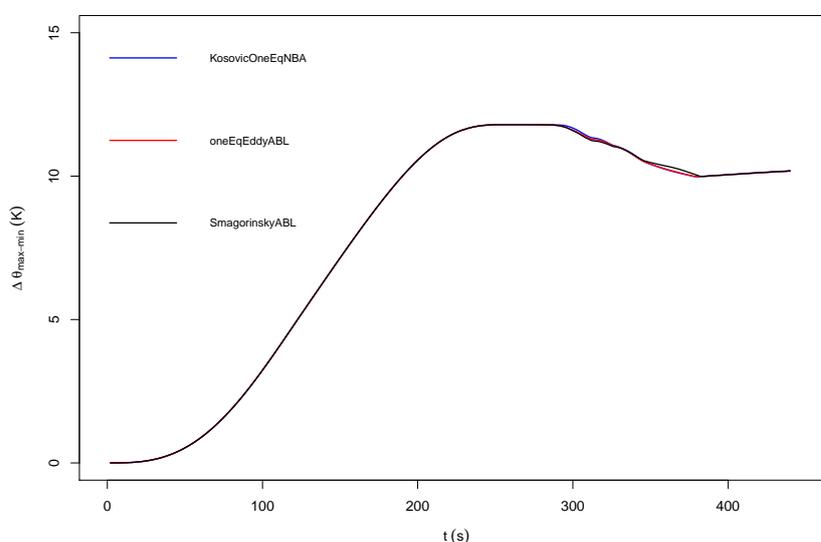
Figura 24 – Campo da temperatura potencial dos diferentes modelos SGS, para o instante $t = 400$ s.



Fonte: Elaborado pelo autor.

Ao implementar a taxa de resfriamento igual a $g(t) = -0,08 \text{ K s}^{-1}$, as diferenças de temperatura máxima e mínima entre a camada de ar fria e o ambiente, não apresentam distinção entre os três modelos ao longo de toda simulação numérica. Esses valores são obtidos a partir de qualquer região do domínio computacional. As diferenças entre essas temperaturas alcançam um valor máximo de aproximadamente 12 K, e posteriormente estabilizam em torno de 10 K (Fig. 25). Esses valores estão de acordo com outros estudos numéricos (ORESKOVIC, 2016; VERMEIRE; ORF; SAVORY, 2011a).

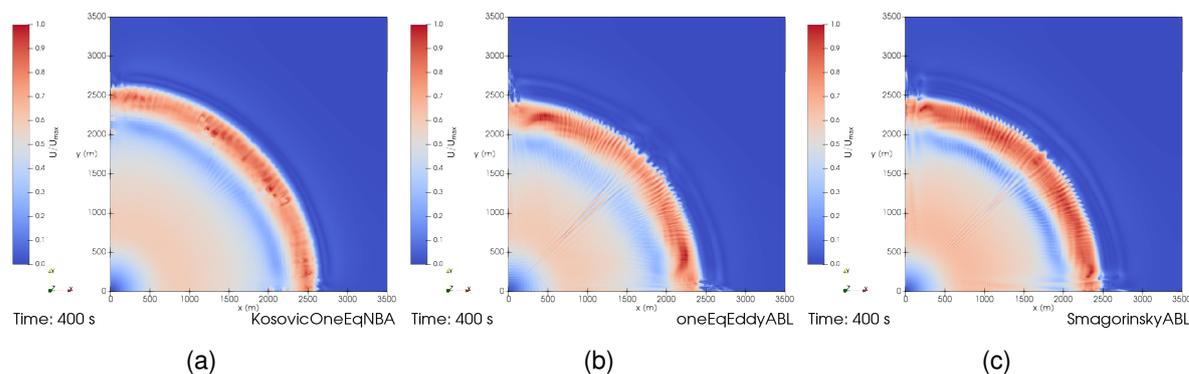
Figura 25 – Evolução temporal da diferença de temperatura potencial máxima e mínima, registrada em qualquer região do domínio computacional.



Fonte: Elaborado pelo autor.

Ao analisar a magnitude do campo da velocidade sobre a superfície, observa-se algumas características do *microburst*. O campo da velocidade é adimensionalizado pelo seu valor máximo, assim é possível comparar os modelos de forma qualitativa (Fig. 26). No instante analisado, a camada de ar fria forma uma região de estagnação sob o jato descendente, seguido da propagação radial da frente de onda. As máximas magnitudes do campo da velocidade localizam-se juntamente com o anel de vórtice. Observa-se que há instabilidades próximo das paredes laterais, provenientes da condição de simetria, influenciando no escoamento próximo dessa região. Devido a essas inconstâncias, as análises e obtenção dos dados nesse trabalho são realizadas no plano diagonal do domínio computacional, representado por 45° no plano x-y.

Figura 26 – Vista superior do plano da superfície, com a magnitude do campo da velocidade adimensional dos diferentes modelos SGS para o instante $t = 400$ s.

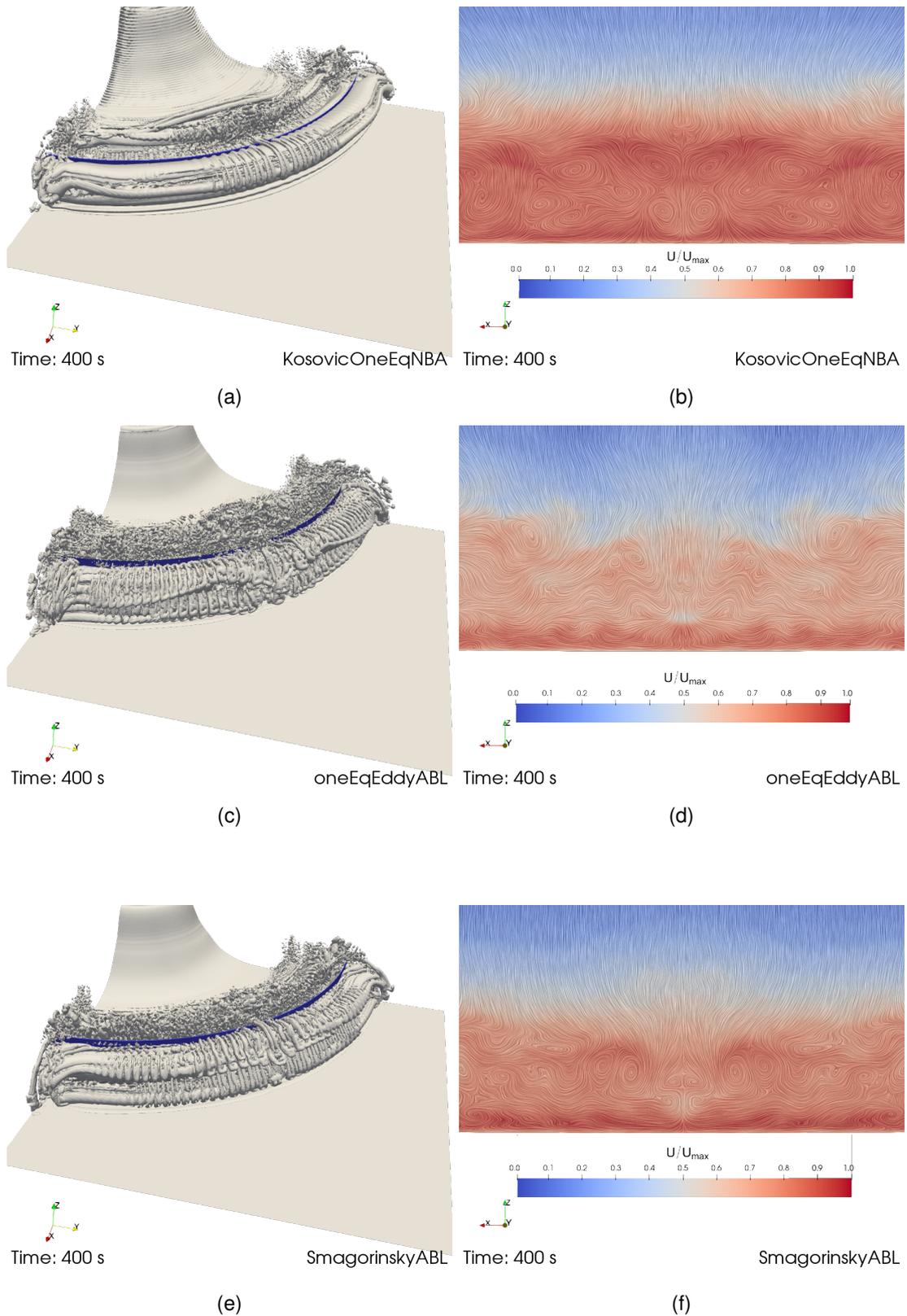


Fonte: Elaborado pelo autor.

Para analisar as instabilidades laterais na frente da onda no instante da Fig. 26, geram-se isosuperfícies do critério Q (Figs. 27a, 27c, 27e), e cortes circulares na frente da onda com o campo da velocidade (Figs. 27b, 27d, 27f). O critério Q é um método de identificação de vórtice (HOLMÉN, 2012). Os cortes circulares são realizados aproximadamente no centro da frente da onda. As faces geradas estão destacadas em azul nas isosuperfícies. As distâncias radiais dos cortes utilizadas nos modelos *KosovicOneEqNBA*, *oneEqEddyABL* e *SmagorinskyABL*, são: 2500 m, 2330 m e 2365 m, respectivamente. Ao analisar as isosuperfícies, os modelos *oneEqEddyABL* (Fig. 27c) e *SmagorinskyABL* (Fig. 27e) apresentam maior difusividade em seu campo do que o modelo *KosovicOneEqNBA* (Fig. 27a) nesse instante analisado. A partir das isosuperfícies também fica mais visível a influência das faces de simetria sobre o escoamento.

Para análise dos cortes circulares, a magnitude do campo da velocidade é adimensionalizada, e linhas de correntes são adicionadas em conjunto. Identificam-se núcleos de vórtices ao longo da frente da onda, sendo mais visíveis nos modelos *KosovicOneEqNBA* e *SmagorinskyABL*. Essa característica possui similaridade com estudos de ondas de densidade cilíndricas. Salinas et al. (2018) observaram em seus estudos que há vórtices *hairpin* para cada lóbulo na frente da onda, onde os autores confirmaram ainda que a observação de vórtices *quasi-streamwise* em estudos anteriores (CANTERO; BALACHANDAR; GARCIA, 2007), são pernas de vórtices *hairpin*.

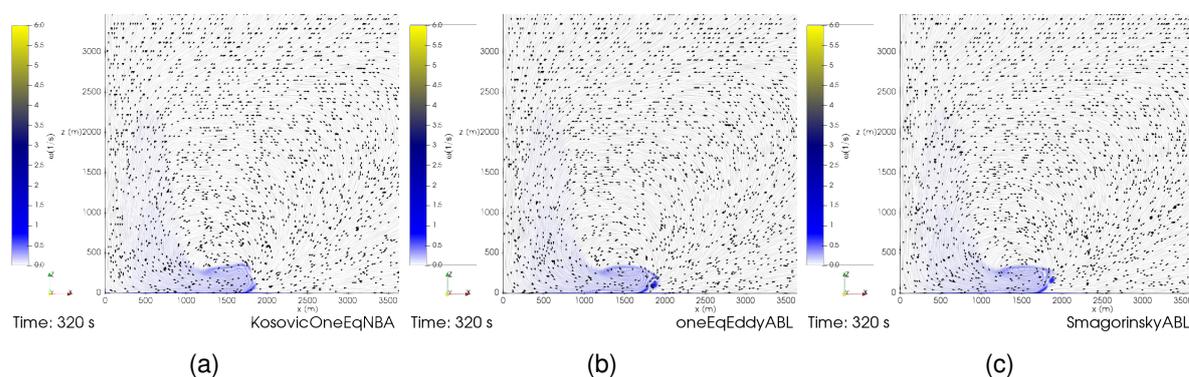
Figura 27 – Isosuperfícies do critério Q para valor de $Q = 0,0001$, e cortes transversais aproximadamente nos centros dos vórtices primários, apresentando o campo de velocidade adimensional para $t = 400$ s.



Fonte: Elaborado pelo autor.

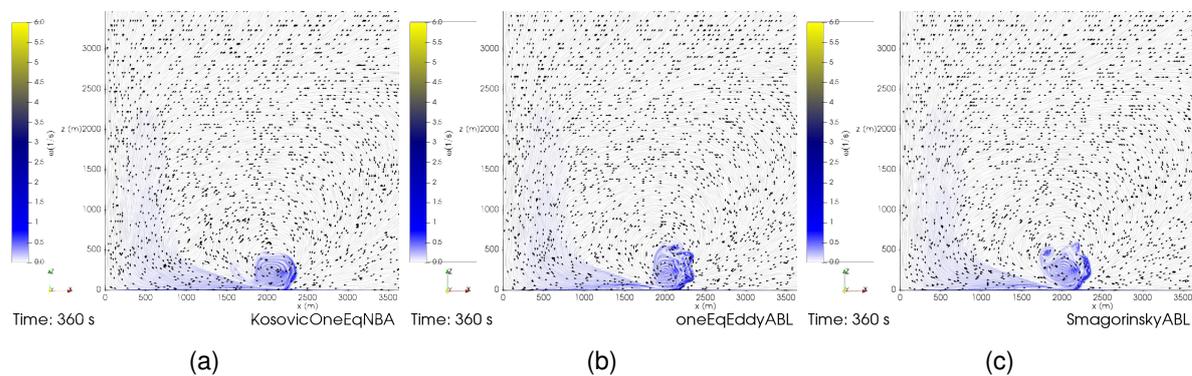
Para que a evolução da propagação seja melhor compreendida, analisa-se o campo da vorticidade para diferentes instantes das simulações numéricas. Linhas de corrente e *glyphs* do campo da velocidade são colocadas em conjunto. Os instantes analisados são: 320 s, 360 s, 400 s e 430 s. A partir desses instantes, torna-se visível o avanço do anel de vórtice rotacionando no sentido anti-horário em todos modelos. Ao analisar o instante 320 s, os modelos *oneEqEddyABL* (Fig. 29b) e *SmagorinskyABL* (Fig. 29c), começam desenvolver um vórtice secundário na frente da onda. Esse vórtice deve-se ao atrito entre o avanço da camada de ar e o solo. Conforme o tempo passa, em 360 s os vórtices secundários deslocam-se sobre o anel de vórtice, elevando a frente da onda (Figs. 30b e 30c). Essa transferência de momento na vertical, faz com que aumente a difusividade da frente da onda, como observado nos instantes subsequentes (Figs. 31b, 31c, 32b e 32c). Já ao observar o modelo de *KosovicOneEqNBA* no instante 400 s, percebe-se que há a aparição de dois vórtices sobrepondo o anel de vórtice (Fig. 31a). Posteriormente, esses vórtices também aumentam a mistura da onda, fazendo com que em 430 s o campo torne-se mais difuso (Fig. 32a).

Figura 29 – Campo da vorticidade dos modelos SGS, para o instante $t = 320$ s. As Linhas de corrente e os *glyphs* são tangentes ao campo da velocidade.



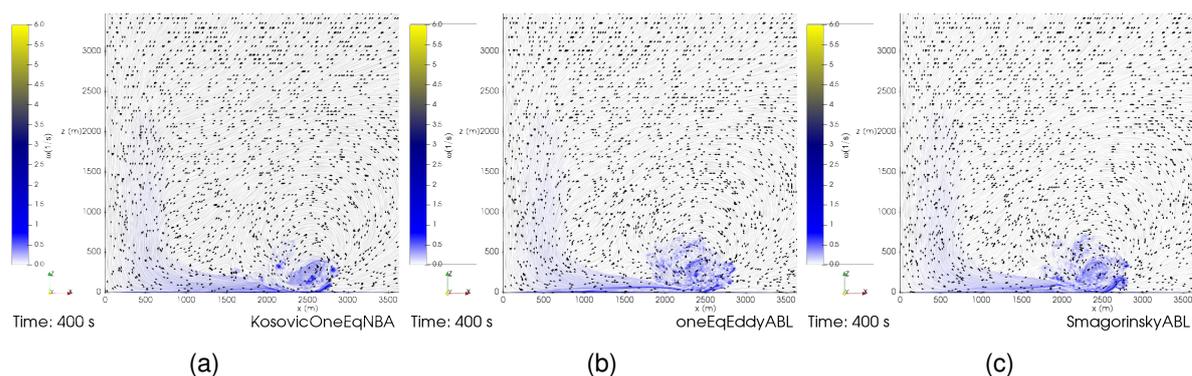
Fonte: Elaborado pelo autor.

Figura 30 – Campo da vorticidade dos modelos SGS, para o instante $t = 360$ s. As Linhas de corrente e os *glyphs* são tangentes ao campo da velocidade.



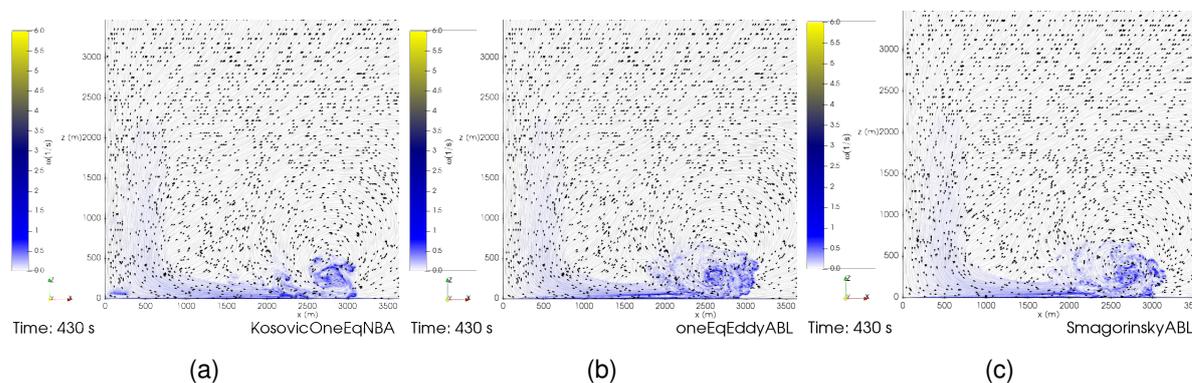
Fonte: Elaborado pelo autor.

Figura 31 – Campo da vorticidade dos modelos SGS, para o instante $t = 400$ s. As Linhas de corrente e os *glyphs* são tangentes ao campo da velocidade.



Fonte: Elaborado pelo autor.

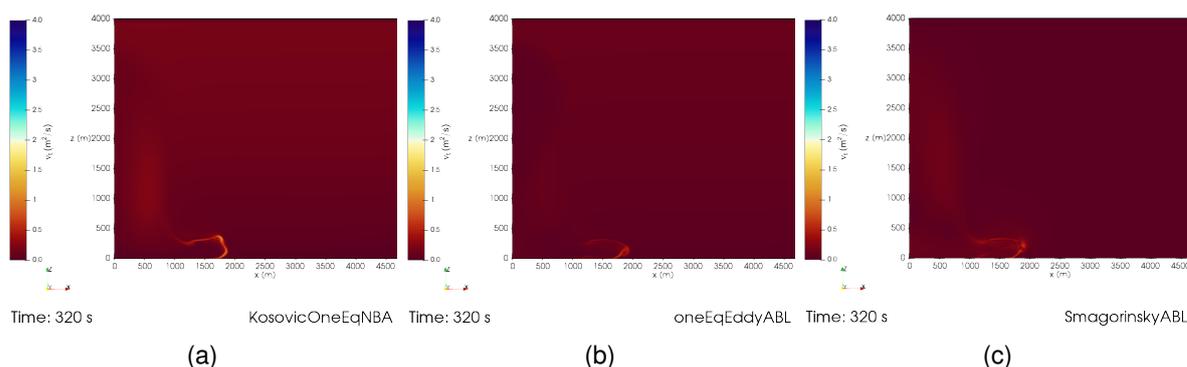
Figura 32 – Campo da vorticidade dos modelos SGS, para o instante $t = 430$ s. As Linhas de corrente e os *glyphs* são tangentes ao campo da velocidade.



Fonte: Elaborado pelo autor.

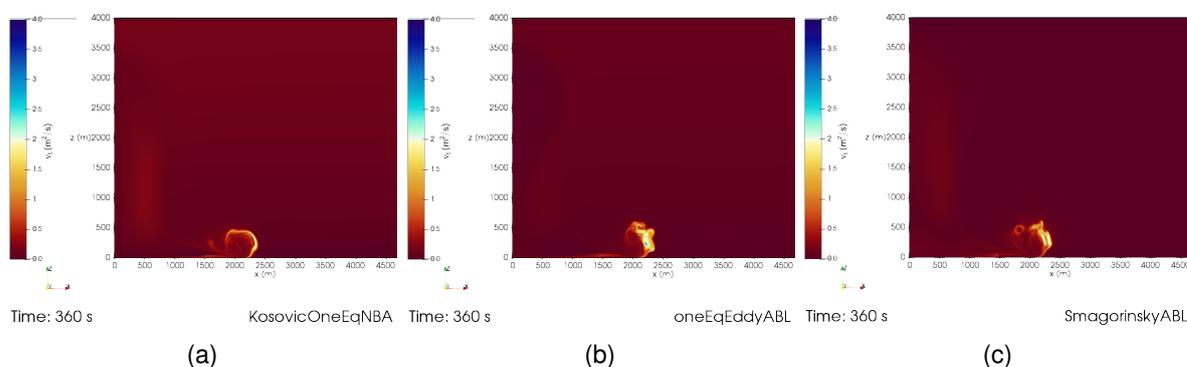
As difusividades nos campos da vorticidade são melhor observadas, ao serem analisados os campos da viscosidade cinemática turbulenta para os mesmos instantes de tempo. Diante dessa variável estar associada ao escoamento, sendo dependente da turbulência. Em 320 s observa-se que o modelo *KosovicOneEqNBA* apresenta uma intensidade ligeiramente maior ao redor do anel de vórtice (Fig. 33a), do que os modelos *oneEqEddyABL* (Fig. 33b) e *SmagorinskyABL* (Fig. 33c), os quais concentram as maiores intensidades no vórtice secundário. Em 360 s o modelo *KosovicOneEqNBA* concentra ainda maior intensidade ao redor do anel de vórtice, apresentando um vórtice primário mais estruturado (Fig. 34a). Já os modelos *oneEqEddyABL* (Fig. 34b) e *SmagorinskyABL* (Fig. 34c), exibem maior mistura na frente da onda nesse instante, devido ao deslocamento dos vórtices secundários sobre ela. Nos tempos subsequentes, há aumento da difusividade em toda frente da onda nos três casos (Figs. 35a, 35b, 35c, 36a, 36b e 36c).

Figura 33 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 320$ s.



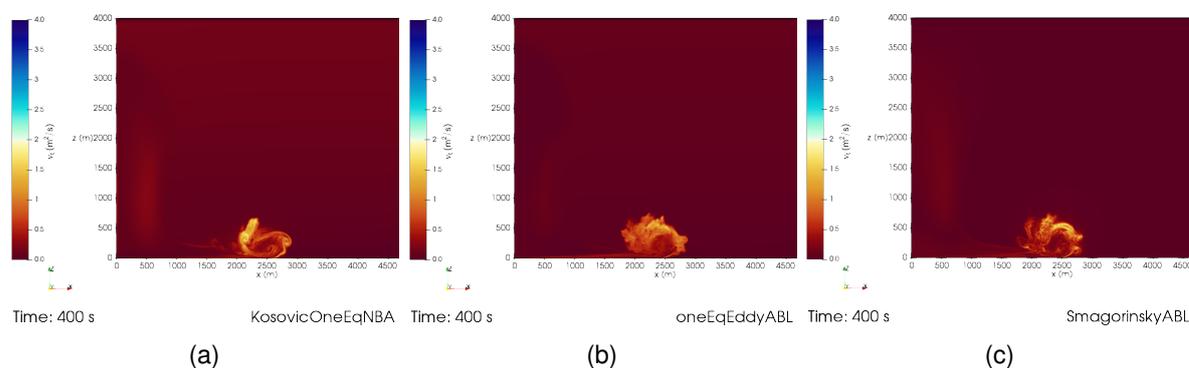
Fonte: Elaborado pelo autor.

Figura 34 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 360$ s.



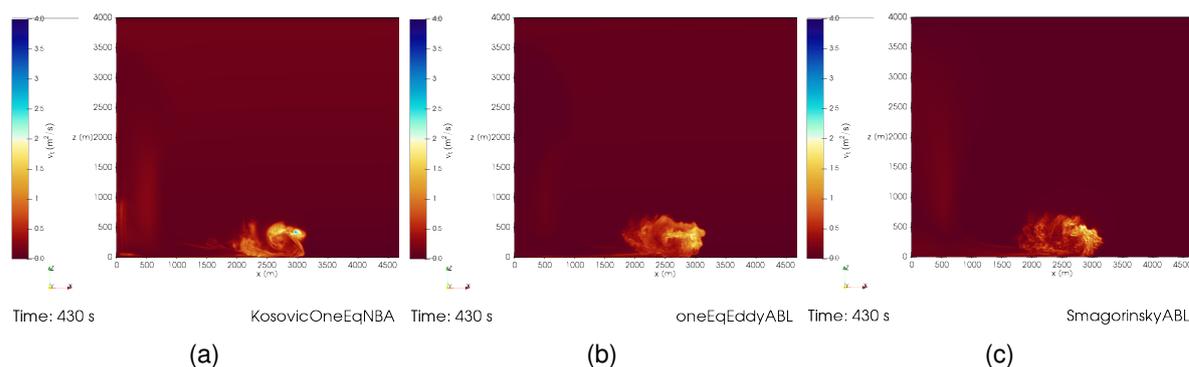
Fonte: Elaborado pelo autor.

Figura 35 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 400$ s.



Fonte: Elaborado pelo autor.

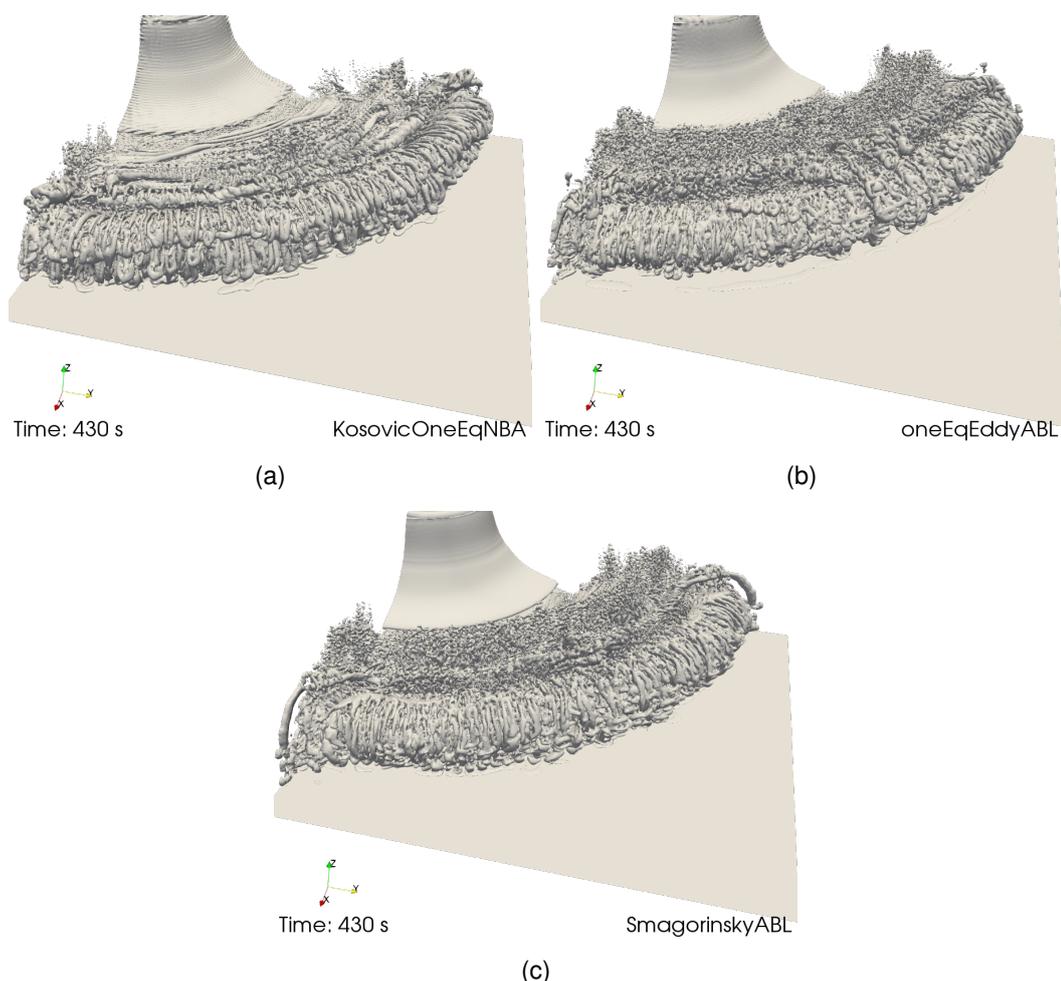
Figura 36 – Campo da viscosidade cinemática turbulenta dos modelos SGS, para o instante $t = 430$ s.



Fonte: Elaborado pelo autor.

Diante dessas análises, observa-se que os três modelos geraram vórtice secundário. Além disso, o aumento da difusividade na frente da onda ocorreu após esses vórtices sobreporem o vórtice primário. Para auxiliar na observação do aumento da turbulência, iso-superfícies do critério Q são gerados para o instante 430 s. Ao comparar as Figs. 27a e 37, houve grande aumento na perturbação da frente da onda do modelo *KosovicOneEqNBA*, proveniente dos vórtices que sobrepuseram o anel de vórtice. Além do mais, observa-se um aumento na mistura dos modelos *oneEqEddyABL* (Fig. 37b) e *SmagorinskyABL* (Fig. 37b), ao comparar com as Figs. 27c e 27e.

Figura 37 – Isosuperfícies do critério Q, para valor de $Q = 0,0001$, no instante $t = 430$ s.

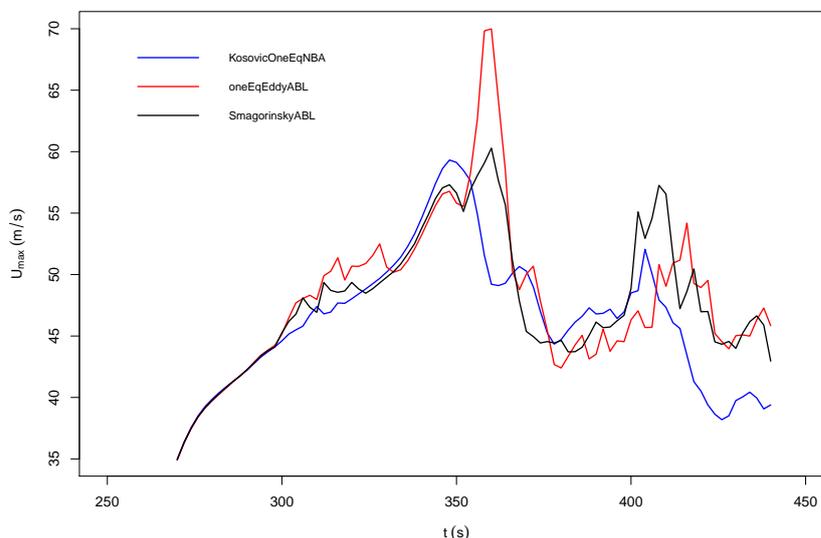


Fonte: Elaborado pelo autor.

As análises realizadas até o momento, consistiram em observar a estrutura da propagação da frente da onda. A partir dessa parte do trabalho serão observadas características da intensidade do vento. Inicialmente observa-se a evolução temporal da máxima magnitude da velocidade, tomando-a em qualquer região do domínio computacional (Fig. 38). Ao observar os dados, a camada de ar entra em contato com o solo por volta de 270 s. Após contato com a superfície, as intensidades do vento começam aumentar gradualmente, até alcançar um pico máximo ($U_{p,max}$), e então começam decair. Contudo, após alguns instantes, há um segundo pico de velocidade em todos modelos, porém, com menor intensidade. O registro do primeiro pico do modelo *KosovicOneEqNBA* ocorre em 348 s, com intensidade da magnitude do vento de aproximadamente 59 m/s. Já para os modelos *oneEqEddyABL* e *SmagorinskyABL*, ambos ocorrem em 360 s, com intensidades de 69 m/s e 60 m/s, respectivamente. Nesses dois casos, as componentes verticais das velocidades possuem elevados valores. No modelo *oneEqEddyABL*, essa intensidade é de 25 m/s, e no *SmagorinskyABL* é igual a 38 m/s. Esses altos valores da componente vertical

estão associados a elevação da frente do anel de vórtice, proveniente do desenvolvimento do vórtice secundário, como observado nas Figs. 30b e 30c.

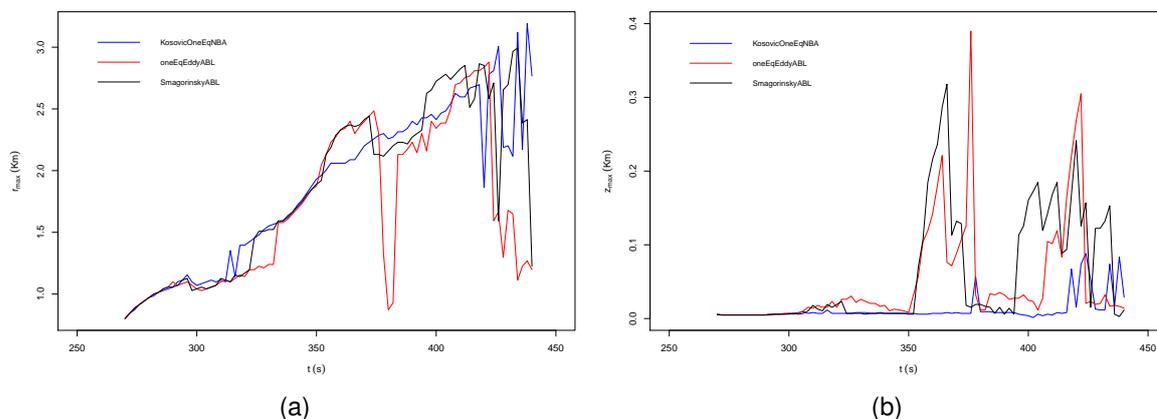
Figura 38 – Evolução temporal dos picos máximos das velocidades nas simulações numéricas, registrados em qualquer região do plano diagonal do domínio computacional.



Fonte: Elaborado pelo autor.

A partir dessas máximas, geram-se as posições radiais e alturas que elas ocorrem em função do tempo (Figs. 39a e 39b). Observa-se que as máximas intensidades distanciam-se cada vez mais do jato descendente com o tempo, propagando-se juntamente com o anel de vórtice. No instante 380 s, a distância radial do modelo *oneEqEddyABL* tem uma queda para em torno de 870 m, porém logo após volta a registrar acima dos 2 Km. Observa-se que esse instante é quando ocorre a velocidade menos intensa após o pico máximo na Fig. 38. A partir dos 420 s os modelos apresentam grande variação nessas distâncias radiais, devido a grande difusividade na frente da onda e perda de intensidade dos ventos. Ao analisar as alturas das intensidades máximas, todos modelos ocorrem em níveis próximos do solo até 350 s, onde após esse instante, as máximas velocidades dos modelos *oneEqEddyABL* e *SmagorinskyABL* começam atingir altitudes elevadas, decorrentes do desenvolvimento do vórtice secundário. Logo após esses picos, os valores voltam a concentrar-se nos níveis mais baixos por alguns instantes, até novamente começarem apresentar grande variação da altura por volta dos 400 s. Já o modelo *KosovicOneEqNBA*, as máximas concentram-se em níveis mais baixos em relação aos outros modelos ao longo de toda simulação numérica, apresentando maiores níveis e variações em torno de 418 s, com os valores atingindo máximas de 88 m.

Figura 39 – Distâncias radiais e alturas das máximas velocidades em função do tempo.



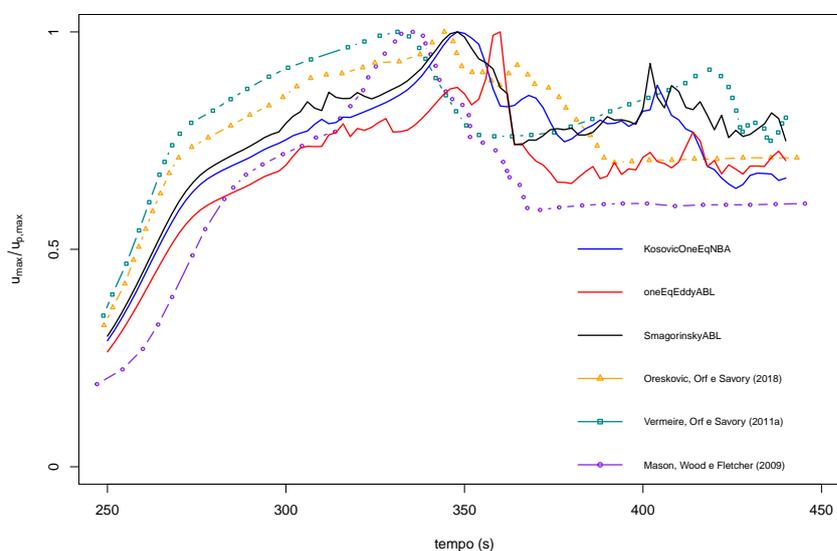
Fonte: Elaborado pelo autor.

Os picos máximos das magnitudes das velocidades registrados nos modelos (Fig. 38), são comparados com outros estudos sob mesmas condições de taxa de resfriamento e dimensões da elipse. No estudo de Vermeire, Orf e Savory (2011a), utilizando o modelo numérico CM1, a máxima registrada foi de 47 m/s. Os diversos casos analisados por Oreskovic, Orf e Savory (2018) na Fig. 19, também eram utilizando o modelo CM1, porém com algumas alterações do trabalho de Vermeire, Orf e Savory (2011a), tais como aumento da ordem de esquema numérico dos termos advectivos para os termos horizontais, e representação completa do microburst. Os casos da Fig. 19 são para $z_o = 0,1$ m, onde a velocidade radial máxima da simulação sob as mesmas condições do termo fonte deste trabalho foi de aproximadamente 50 m/s. Já no caso de Mason, Wood e Fletcher (2009), a velocidade máxima foi de 57 m/s. De modo que, Vermeire, Orf e Savory (2011a) e Oreskovic, Orf e Savory (2018) salientam que o modelo URANS utilizado por Mason, Wood e Fletcher (2009), tende a superestimar a energia cinética turbulenta na região da superfície do escoamento do jato, quando comparado com o modelo LES, resultando em diferenças nas análises.

A distância radial do pico máximo da magnitude da velocidade do modelo KosovicOneEqNBA, ocorreu em 1876 m (Fig. 39a), na altura de 7 m (Fig. 39b). Já a distância radial dos modelos *oneEqEddyABL* e *SmagorinskyABL*, aconteceram na mesma distância radial de 2329 m, nas alturas de 142 m e 216 m, respectivamente. Dentre os estudos numéricos sob mesmas condições de taxa de resfriamento e da elipse, o pico máximo da velocidade de Mason, Wood e Fletcher (2009) para $z_o = 0,02$ m, ocorreu aproximadamente na altura de 11 m e distância radial de 1250 m. No caso de Vermeire, Orf e Savory (2011a) com rugosidade da superfície de $z_o = 0,01$ m, o pico máximo aconteceu na altura de 15 m e localização radial de 1425 m. Enquanto que no estudo de Oreskovic, Orf e Savory (2018) utilizando $z_o = 0,1$ m, a distância e altura da máxima velocidade radial foram aproximadamente de 1930 m e 149 m, respectivamente.

A evolução temporal das velocidades radiais máximas é comparada com outros estudos, onde essas são adimensionalizadas em função do pico máximo da velocidade radial ($u_{p,max}$) (Fig. 40). As evoluções dos modelos apresentam algumas diferenças da Fig. 38, diante das velocidades radiais máximas serem relacionadas por $u_{max} = \sqrt{u_x^2 + u_y^2}$. Os instantes dos $u_{p,max}$ dos modelos *oneEqEddyABL* e *KosovicOneEqNBA*, mantiveram os tempos registrados dos picos máximos das magnitudes (Fig. 38), contudo o instante do modelo *SmagorinskyABL* passou a ocorrer em 348 s. Ao comparar as evoluções com outros estudos, observa-se que os instantes dos picos máximos são próximos. Após 370 s, as máximas mantêm-se quase constantes ao longo do tempo no estudo de Mason, Wood e Fletcher (2009), onde os autores associam essa característica como as intensidades voltando a concentrarem-se na região de impacto, e não mais na estrutura frontal. Mesma característica é observada no estudo de Oreskovic, Orf e Savory (2018), porém acontecendo por volta dos 390 s. Como observado na Fig. 39a, as máximas de todos modelos SGS até 422 s ainda encontram-se em grandes distâncias radiais, o que resulta em uma variabilidade desses valores após o pico máximo na Fig. 40, diante da grande difusividade na frente da onda.

Figura 40 – Evolução das máximas velocidades radiais em função do tempo, adimensionalizadas pelos picos radiais máximos dos ventos.

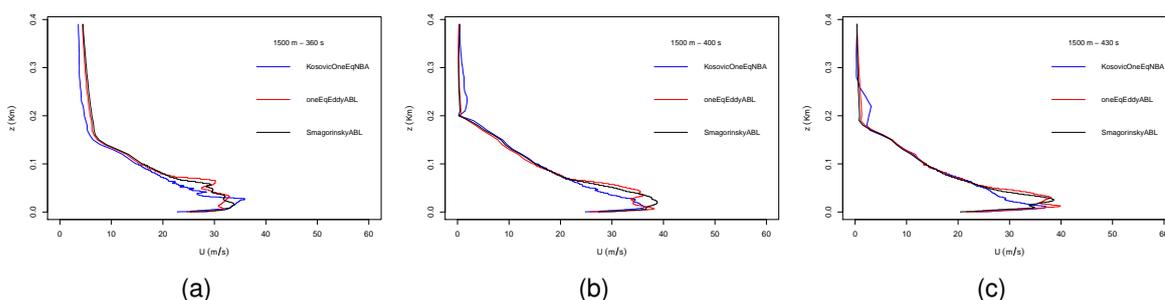


Fonte: Elaborado pelo autor.

Perfis verticais da magnitude do vento são gerados para diferentes posições radiais. Os pontos analisados são em 1500 m e 2000 m, para os instantes: 360 s, 400 s e 430 s. Dentre as características a serem levadas em consideração nesses pontos, é que nesses instantes as frentes das ondas já transpassaram os 1500 m, e os centros das ondas encontram-se aproximadamente na posição de 2000 m em 360 s (Figs. 30a, 30b e 30c). Observando os perfis em 1500 m (Fig. 41), percebe-se que há pouca variação entre

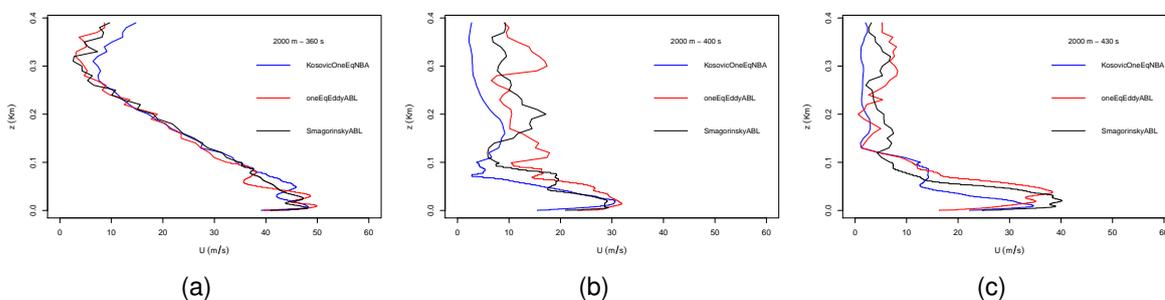
os perfis nos três casos. Isso ocorre devido a ser uma região com pouca interferência da frente da onda, onde o aumento da difusividade ocorre em distâncias maiores. Há um aumento nas intensidades do vento próximo da superfície conforme o tempo passa. O modelo *KosovicOneEqNBA* apresenta ventos intensos em níveis menores do que os outros modelos. Na posição 2000 m, os perfis dos modelos apresentam intensidades muito similares em 360 s (Fig. 42a). Nesse instante o vórtice secundário desloca-se sobre o primário nos modelos *oneEqEddyABL* e *SmagorinskyABL*, e as frentes das ondas ainda mantêm-se estruturadas (Figs. 30b e 30c). Em 400 s aumenta-se significativamente a difusividade desses dois modelos, o que resulta em maiores diferenças entre os perfis em níveis mais elevados. Outra característica nesse instante é que há uma queda brusca da velocidade próximo ao solo (Fig. 42b). Já em 430 s, a frente da onda encontra-se em distâncias maiores que 2000 m, dessa maneira as intensidades aproximam-se novamente em níveis elevados. Na região próximo à superfície, o modelo de *KosovicOneEqNBA* ainda apresenta ventos intensos em níveis mais baixos do que os outros modelos (Fig. 42c).

Figura 41 – Perfis da velocidade do vento na distância radial de 1500 m, nos instantes: 360 s, 400 s e 430 s.



Fonte: Elaborado pelo autor.

Figura 42 – Perfis da velocidade do vento na distância radial de 2000 m, nos instantes: 360 s, 400 s e 430 s.

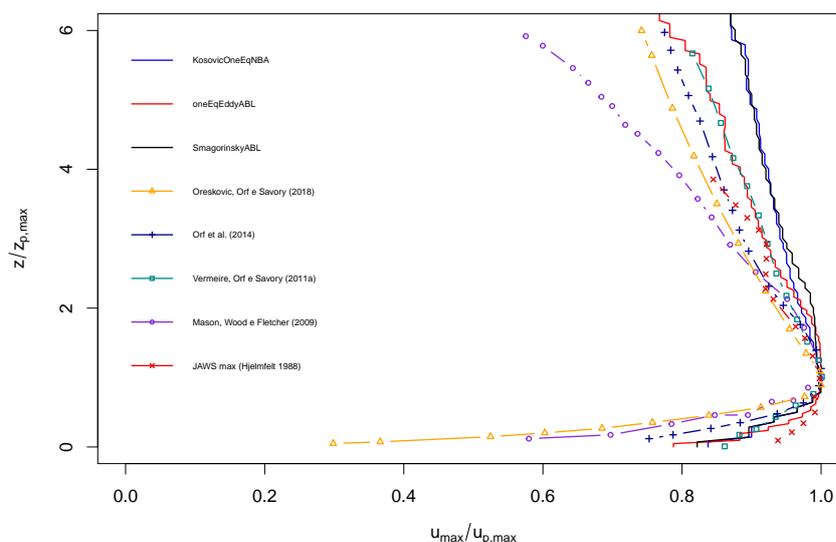


Fonte: Elaborado pelo autor.

Para que os picos máximos das velocidades dos modelos sejam comparados com outros estudos, perfis verticais das velocidades radiais são gerados, próximo das regiões

onde as máximas acontecem (Fig. 43). Os perfis dos modelos são comparados com estudos similares ((VERMEIRE; ORF; SAVORY, 2011a; MASON; WOOD; FLETCHER, 2009; ORESKOVIC; ORF; SAVORY, 2018)), com um caso de modelo de nuvem completa (ORF et al., 2014), e um perfil máximo registrado em dados de campo do projeto JAWS (HJELMFELT, 1988). As velocidades são adimensionalizadas em função do pico máximo da velocidade radial, e a altura em relação ao nível que esse máximo ocorre. No modelo *oneEqEddyABL* apesar do caso apresentar a velocidade radial máxima no instante 360 s, na altura de 142 m, essa localização encontra-se na inclinação da frente da onda, a qual é ausente de ventos intensos nos níveis inferiores, então decide-se tomar o perfil para o pico máximo da velocidade radial no instante 348 s. Com isso, as alturas dos picos máximos das velocidades radiais dos modelos *KosovicOneEqNBA* e *SmagorinskyABL*, acontecem em 7 m, e no modelo *oneEqEddyABL* em 10,4 m. Os perfis dos modelos *KosovicOneEqNBA* e *SmagorinskyABL* apresentam bastante similaridade. Ao comparar os perfis acima do nível do pico máximo, o modelo *oneEqEddyABL* possui maior proximidade dos outros estudos, apresentando grande semelhança com o estudo de Vermeire, Orf e Savory (2011a). Na região inferior do pico máximo, o modelo *oneEqEddyABL* também aproximou-se mais do perfil registrado no projeto JAWS. Entretanto, como salientado por Orf et al. (2014), esses tipos de gráficos adimensionalizados devem ser analisados com cuidado, onde pequenas diferenças nos perfis podem resultar em significativas diferenças de carga do vento. Os autores descrevem que para análises de carga do vento sobre estruturas da engenharia é necessário analisar não somente o local do pico máximo, mas também ao redor e em todas alturas abaixo desse ponto.

Figura 43 – Perfis adimensionais da velocidade radial do vento, próximo da região do pico máximo.



Fonte: Elaborado pelo autor.

5 CONSIDERAÇÕES FINAIS

O objetivo inicial do presente trabalho era de desenvolver um código numérico no software OpenFOAM utilizando a metodologia de fonte de resfriamento elaborada por Anderson, Orf e Straka (1992). O desenvolvimento do código foi alcançado com êxito, onde posteriormente a segunda proposta do trabalho era averiguar a sensibilidade de diferentes parametrizações de turbulência sobre o efluxo de microbursts. Os modelos utilizados fazem parte do conjunto de ferramentas do SOWFA, sendo dois modelos lineares (*one-EqEddyABL* e *SmagorinskyABL*) e um não-linear (*KosovicOneEqNBA*). Dentre as diferenças dos modelos, está que o modelo não-linear consegue reproduzir as características anisotrópica e de retroespalhamento de um escoamento induzido por cisalhamento. As simulações demonstram de forma objetiva a sensibilidade da mudança das parametrizações sobre a propagação da frente de onda, e como isso favorece a formação e manutenção desses vórtices. Tais resultados mostram novas possibilidades de estudos que permitem respostas de análises científicas adicionais.

Além disso, ao analisar a equação do modelo de fonte de resfriamento, uma importante correção é identificada em sua descrição apresentada por trabalhos anteriores. Primeiramente, Oreskovic, Orf e Savory (2018) mencionam que há um erro de texto na consideração do limite das bordas da elipse em estudos anteriores. Então no presente trabalho constata-se que além dessa correção, mais uma manipulação adicional deve ser realizada no termo espacial (Eq. 26), de forma que o resfriamento seja de acordo com o modelo. Os resultados obtidos após implementação do modelo são promissores, os quais reproduzem as características de um *microburst* de maneira similar com estudos anteriores.

Pelo fato das simulações realizadas serem para um *microburst* estático, os casos foram representados por 1/4 do fenômeno para reduzir tempo computacional. Ao considerar a condição de simetria nas paredes, observou-se que essas tiveram influência sobre o escoamento próximo a essa região, não conseguindo reproduzir adequadamente a simetria. Dessa maneira as análises foram realizadas no plano diagonal do domínio computacional. Dentre as características observadas na propagação dos *microbursts* dos diferentes casos, foram:

1. Todos modelos formam vórtices secundários provenientes do atrito com a superfície. Os modelos lineares desenvolvem em tempos similares, em torno dos 320 s. Já o modelo não-linear, após aproximadamente 360 s.
2. Os vórtices secundários movem-se ao longo dos vórtices primários após serem desenvolvidos junto a superfície. Poucos instantes após, há um grande aumento de difusividade na frente da onda.
3. A inclinação da frente de onda nos modelos lineares, proveniente do vórtice secun-

dário, fez com que os picos máximos da magnitude da velocidade ao longo do tempo, atingissem elevadas altitudes, ultrapassando os 300 m em alguns instantes. Já no modelo não-linear houve pouca variação nas alturas dos picos máximos, mesmo após o desenvolvimento do vórtice secundário. As alturas desse modelo atingiram níveis mais altos após aumento da difusividade na frente da onda, mas não ultrapassando os 90 m.

4. As intensidades máximas das velocidades propagaram-se para distâncias radiais maiores no decorrer do tempo, até começarem apresentar grande variação da localização radial em torno dos 420 s. Por volta dos 380 s houve uma queda brusca desse valor radial no modelo *oneEqEddyABL*, sendo um momento que coincidiu com uma queda da máxima velocidade. Contudo, logo após o modelo voltou a registrar as máximas em elevadas distâncias radiais.
5. As máximas velocidades registradas pegando os valores em qualquer região do domínio computacional, aumentaram os seus valores gradualmente ao longo do tempo após contato com a superfície até alcançarem um pico máximo. O instante do pico máximo da magnitude da velocidade do modelo não-linear ocorreu em 348 s, e nos modelos lineares coincidiram em 360 s. Já ao analisar os picos máximos das velocidades radiais, os modelos *KosovicOneEqNBA* e *oneEqEddyABL* mantiveram os instantes observados nas máximas magnitudes, contudo o instante do modelo *SmagorinskyABL* passou a ocorrer em 348 s. Essa diferença deve-se pelo fato do modelo possuir uma elevada componente vertical da velocidade em 360 s.
6. Ao comparar os perfis verticais da velocidade radial na região do pico máximo, o modelo *oneEqEddyABL* apresentou maior proximidade com outros estudos numéricos da literatura e com um perfil vertical registrado no projeto JAWS. Contudo, salienta-se que diferente dos outros modelos utilizados neste trabalho, o perfil gerado do modelo *oneEqEddyABL* não foi no instante do pico máximo da velocidade radial. Essa escolha deveu-se porque o pico máximo ocorreu em um elevado nível na inclinação da frente da onda, ausente de ventos intensos nos níveis inferiores. Dessa maneira, decidiu-se gerar o perfil instantes antes, em 348 s, onde a máxima localizava-se na região inferior do anel de vórtice.

Os resultados obtidos são importantes para ajudar na compreensão do fenômeno, onde o código desenvolvido no software OpenFOAM mostrou-se uma excelente ferramenta para realizar tais análises. Os modelos lineares apresentaram similaridades na estrutura e desenvolvimento da frente de onda, e também na ocorrência das máximas velocidades em elevadas altitudes. O modelo não-linear apresentou menor difusividade na frente da onda durante a propagação do que os modelos lineares, também registrando as máximas

velocidades em níveis menores, onde com o decorrer do tempo todos modelos apresentaram uma grande mistura na frente da onda. Diante disso, observa-se que há influência da parametrização de turbulência na propagação do fenômeno, onde os diferentes modelos SGS modificaram os padrões obtidos na análise das grandezas do escoamento.

Tratando-se de um fenômeno com curto tempo de vida e escala espacial pequena, há uma grande dificuldade observacional, existindo poucos exemplos de microbursts com dados de vento em alta frequência temporal e em diferentes níveis. Dessa maneira, a escolha do modelo mais adequado para reproduzir as características do efluxo do microburst seria presunçosa, necessitando de dados observacionais para uma maior validação. Contudo, espera-se que o modelo *KosovicOneEqNBA* reproduza melhor as características de propagação do fenômeno, devido a turbulência do escoamento sobre a superfície ser anisotrópica. As análises realizadas por KOSOVIĆ (1997) foram sobre condições similares a deste estudo, utilizando LES em um escoamento induzido por cisalhamento com alto número de Reynolds. Os resultados mostraram que o modelo não-linear reproduz melhor as características do escoamento quando comparado com um modelo linear, apresentando maior proximidade com a predição da teoria de Monin-Obukhov e resultados experimentais. Dentre as razões descritas para essa melhora, está na redistribuição significativa da energia cinética turbulenta entre as componentes de tensão normal da subgrade.

A partir das análises realizadas, para tornar o caso mais próximo de um ambiente real, visa-se adicionar ventos geostróficos e perturbação no termo fonte em trabalhos futuros. Diante do registro de casos reais apresentarem uma estrutura mais complexa (HJELMFELT, 1988), assim como observado em estudos com modelo de nuvem completa (ORF; KANTOR; SAVORY, 2012; ORF et al., 2014). Ademais, sugere-se que seja realizada validação de malha do caso, a qual foi baseada em outros estudos, para observar se os picos máximos da velocidade se manterão, principalmente no modelo *oneEqEddyABL*, que apresentou valores mais elevados e discrepantes em relação aos outros modelos. Além disso, como observado a condição de simetria não consegue reproduzir adequadamente o escoamento próximo às paredes, então sugere-se que a simulação analise um caso completo do fenômeno.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABBÀ, A.; CERCIGNANI, A.; VALDETTARO, L. Analysis of subgrid scale models. **Computers & Mathematics with Applications**, Elsevier, v. 46, n. 4, p. 521–535, 2003.
- ABD-ELAAL, E.-S.; MILLS, J. E.; MA, X. Numerical simulation of downburst wind flow over real topography. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 172, p. 85–95, 2018.
- ABKAR, M.; PORTÉ-AGEL, F. A new boundary condition for large-eddy simulation of boundary-layer flow over surface roughness transitions. **Journal of Turbulence**, Taylor & Francis, n. 13, p. N23, 2012.
- ABOSHOSHA, H.; DAMATTY, A. E. Engineering method for estimating the reactions of transmission line conductors under downburst winds. **Engineering Structures**, Elsevier, v. 99, p. 272–284, 2015.
- ANABOR, V. et al. Large-eddy simulation of a microburst. **Atmospheric Chemistry and Physics**, Copernicus GmbH, v. 11, n. 17, p. 9323–9331, 2011.
- ANDERSON, J.; ORF, L.; STRAKA, J. A 3-d model system for simulating thunderstorm microburst outflows. **Meteorology and Atmospheric Physics**, Springer, v. 49, n. 1-4, p. 125–131, 1992.
- ANDERSSON, B. et al. **Computational fluid dynamics for engineers**. [S.l.]: Cambridge University Press, 2011.
- ARYA, P. S. **Introduction to micrometeorology**. [S.l.]: Elsevier, 2001.
- BRYAN, G. H.; FRITSCH, J. M. A benchmark simulation for moist nonhydrostatic numerical models. **Monthly Weather Review**, v. 130, n. 12, p. 2917–2928, 2002.
- BURLANDO, M. et al. Investigation of the weather conditions during the collapse of the morandi bridge in genoa on 14 august 2018 using field observations and wrf model. **Atmosphere**, Multidisciplinary Digital Publishing Institute, v. 11, n. 7, p. 724, 2020.
- BYERS, H. R.; BRAHAM, R. R. **The thunderstorm: report of the Thunderstorm Project**. [S.l.]: US Government Printing Office, 1949.
- CANTERO, M. I.; BALACHANDAR, S.; GARCIA, M. H. High-resolution simulations of cylindrical density currents. **Journal of Fluid Mechanics**, Cambridge University Press, v. 590, p. 437–469, 2007.
- CHAY, M.; ALBERMANI, F.; WILSON, R. Numerical and analytical simulation of downburst wind loads. **Engineering Structures**, Elsevier, v. 28, n. 2, p. 240–254, 2006.
- CHURCHFIELD, M. **Simulator for Wind Farm Applications (SOWFA)**. 2017. <https://www.aere.iastate.edu/nawea2017/files/2017/09/SOWFA_Training_NAWEA_2017.pdf>. Accessed: 15 aug. 2020.

DAMATY, A. E.; ELAWADY, A. Critical load cases for lattice transmission line structures subjected to downbursts: Economic implications for design of transmission lines. **Engineering structures**, Elsevier, v. 159, p. 213–226, 2018.

DAS, K. K.; GHOSH, A.; SINHAMAHAPATRA, K. Experimental and numerical simulation of the translational downburst using impinging jet model. **Int. J. Eng. Sci. Technol.(IJEST)**, v. 3, p. 4656–4667, 2011.

DAVIDSON, P. A. **Turbulence: an introduction for scientists and engineers**. [S.I.]: Oxford University Press, 2015.

DEARDORFF, J. W. Stratocumulus-capped mixed layers derived from a three-dimensional model. **Boundary-Layer Meteorology**, Springer, v. 18, n. 4, p. 495–527, 1980.

DEMARCO, G. et al. Combined effect of jet impingement and density perturbation forcing on the evolution of laboratory-simulated microbursts. **Journal of wind engineering and industrial aerodynamics**, Elsevier, v. 123, p. 69–76, 2013.

DOMARADZKI, J. A.; LIU, W.; BRACHET, M. E. An analysis of subgrid-scale interactions in numerically simulated isotropic turbulence. **Physics of Fluids A: Fluid Dynamics**, American Institute of Physics, v. 5, n. 7, p. 1747–1759, 1993.

DRIEST, E. R. V. On turbulent flow near a wall. **Journal of the aeronautical sciences**, v. 23, n. 11, p. 1007–1011, 1956.

DURRAN, D. R.; KLEMP, J. B. On the effects of moisture on the brunt-väisälä frequency. **Journal of the Atmospheric Sciences**, v. 39, n. 10, p. 2152–2158, 1982.

ELAWADY, A. et al. Aero-elastic testing of multi-spanned transmission line subjected to downbursts. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 169, p. 194–216, 2017.

FERZIGER, J. H.; PERIĆ, M. **Computational methods for fluid dynamics**. [S.I.]: Springer, 2002. v. 3.

FUJITA, T.; WAKIMOTO, R. Microbursts in jaws depicted by doppler radars, pam, and aerial photographs. 1983.

FUJITA, T. T. Objectives, operations, and results of project nimrod. **Preprints, 11th Conf. on Severe Local Storms**, Kansas City, Amer. Meteor. Soc., p. 259–266, 1979.

_____. Tornadoes and downbursts in the context of generalized planetary scales. **Journal of the Atmospheric Sciences**, v. 38, n. 8, p. 1511–1534, 1981.

_____. The downburst: Microburst and macroburst. **SMRP Research Paper**, University of Chicago, v. 210, p. 122, 1985.

_____. Downbursts: meteorological features and wind field characteristics. **Journal of wind engineering and industrial aerodynamics**, Elsevier, v. 36, p. 75–86, 1990.

FUJITA, T. T.; BYERS, H. R. Spearhead echo and downburst in the crash of an airliner. **Monthly Weather Review**, v. 105, n. 2, p. 129–146, 1977.

GONZATTO, M. **Em capítulos, os motivos do fracasso de um investimento de R\$ 300 milhões**. 2016. <<https://gauchazh.clicrbs.com.br/geral/noticia/2016/06/em-capitulos-os-motivos-do-fracasso-de-um-investimento-de-r-300-milhoes-5939096.html>>. Accessed: 26 fev. 2020.

GRAY, D. D.; GIORGINI, A. The validity of the boussinesq approximation for liquids and gases. **International Journal of Heat and Mass Transfer**, Elsevier, v. 19, n. 5, p. 545–551, 1976.

GRÖTZBACH, G. Direct numerical and large eddy simulations of turbulent channel flows. **Encyclopedia of Fluid Mechanics, Complex Flow Phenomena and Modeling**, Gulf Publishing, 1987, v. 6, p. 1337–1391, 1987.

HAINES, M. R. **The simulation of non-synoptic effects and their implications for engineering structures**. 2015. Tese (Doutorado) — University of Birmingham, 2015.

HJELMFELT, M. et al. Observational and numerical study of a microburst line-producing storm. **Journal of the Atmospheric Sciences**, v. 46, n. 17, p. 2731–2744, 1989.

HJELMFELT, M. R. Structure and life cycle of microburst outflows observed in colorado. **Journal of Applied Meteorology**, v. 27, n. 8, p. 900–927, 1988.

HOLMÉN, V. Methods for vortex identification. **Master's Theses in Mathematical Sciences**, 2012.

HOLMES, J. D. **Wind loading of structures**, ed. TF e-Library., ed. [S.l.]: Taylor & Francis e-Library, 2004.

HOLZMANN, T. Mathematics, numerics, derivations and openfoam®. **Loeben, Germany: Holzmann CFD, URI: <https://holzmann-cfd.de> (visited on Nov. 29, 2017)**, 2016.

HYDRO, M. **A History of Electric Power in Manitoba**. [S.l.]: Manitoba Hydro, 2010.

IRRENFRIED, C.; STEINER, H. Subgrid-scale modeling of turbulent heat transport in forced convection at high molecular prandtl numbers. HEFAT, 2017.

KIM, J.; HANGAN, H. Numerical simulations of impinging jets with application to downbursts. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 95, n. 4, p. 279–298, 2007.

KNUPP, K. R. Numerical simulation of low-level downdraft initiation within precipitating cumulonimbi: Some preliminary results. **Monthly weather review**, v. 117, n. 7, p. 1517–1529, 1989.

KOSOVIĆ, B. Subgrid-scale modelling for the large-eddy simulation of high-reynolds-number boundary layers. **Journal of Fluid Mechanics**, Cambridge University Press, v. 336, p. 151–182, 1997.

KOSOVIĆ, B.; CURRY, J. A. A large eddy simulation study of a quasi-steady, stably stratified atmospheric boundary layer. **Journal of the atmospheric sciences**, v. 57, n. 8, p. 1052–1068, 2000.

KRIDER, E. P. **Thunderstorm**. 2016. <<https://www.britannica.com/science/thunderstorm>>. Accessed: 10 aug. 2020.

KUNDU, P. K.; COHEN, I. M.; DOWLING, D. **Fluid Mechanics 4th**. [S.l.]: Elsevier, 2008.

LALAS, D. P.; RATTO, C. F. **Modelling of atmospheric flow fields**. [S.l.]: World scientific, 1996.

LANDRETH, C. C.; ADRIAN, R. J. Impingement of a low reynolds number turbulent circular jet onto a flat plate at normal incidence. **Experiments in Fluids**, Springer, v. 9, n. 1-2, p. 74–84, 1990.

LEE, M.; MOSER, R. D. Direct numerical simulation of turbulent channel flow up to $Re_\tau \approx 5200$. **Journal of Fluid Mechanics**, Cambridge University Press, v. 774, p. 395–415, 2015.

LI, C. et al. A revised empirical model and cfd simulations for 3d axisymmetric steady-state flows of downbursts and impinging jets. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 102, p. 48–60, 2012.

LILLY, K. The representation of small-scale turbulence in numerical simulation experiments. 1966.

LIMA, E. G. de; LOREDO-SOUZA, A. M. Análise da ocorrência de downbursts no brasil. **Ciência e Natura**, Universidade Federal de Santa Maria, v. 37, n. 1, p. 32–38, 2015.

LIU, F. A thorough description of how wall functions are implemented in openfoam. **Proceedings of CFD with OpenSource Software**, p. 1–33, 2016.

MALALASEKERA, W.; VERSTEEG, H. An introduction to computational fluid dynamics. **The finite volume method**, Harlow: Prentice Hall, p. 1995, 2007.

MASON, M. S. Simulation of downburst wind fields. University of Sydney, 2009.

MASON, M. S.; FLETCHER, D.; WOOD, G. Numerical simulation of idealised three-dimensional downburst wind fields. **Engineering Structures**, Elsevier, v. 32, n. 11, p. 3558–3570, 2010.

MASON, M. S.; WOOD, G. S. Influence of jet inclination on structural loading in an experimentally simulated microburst. 2005.

MASON, M. S.; WOOD, G. S.; FLETCHER, D. F. Numerical simulation of downburst winds. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 97, n. 11-12, p. 523–539, 2009.

MOENG, C.-H. A large-eddy-simulation model for the study of planetary boundary-layer turbulence. **Journal of the Atmospheric Sciences**, v. 41, n. 13, p. 2052–2062, 1984.

MOENG, C.-H.; WYNGAARD, J. C. Spectral analysis of large-eddy simulations of the convective boundary layer. **Journal of the Atmospheric Sciences**, v. 45, n. 23, p. 3573–3587, 1988.

NGUYEN, H. H.; MANUEL, L. Transient thunderstorm downbursts and their effects on wind turbines. **Energies**, Multidisciplinary Digital Publishing Institute, v. 7, n. 10, p. 6527–6548, 2014.

NGUYEN, H. H. et al. Simulation of thunderstorm downbursts and associated wind turbine loads. **Journal of solar energy engineering**, American Society of Mechanical Engineers Digital Collection, v. 135, n. 2, 2013.

OPENCFD. Openfoam: User guide v2006. **OpenFOAM v2006**, URI: <https://www.openfoam.com/documentation/guides/latest/doc/index.html> (visited on Nov. 24, 2020), 2020.

ORESKOVIC, C. Numerical investigation of full scale thunderstorm downbursts: A parametric study and comparison to meteorological model. 2016.

ORESKOVIC, C.; ORF, L. G.; SAVORY, E. A parametric study of downbursts using a full-scale cooling source model. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 180, p. 168–181, 2018.

ORF, L.; KANTOR, E.; SAVORY, E. Simulation of a downburst-producing thunderstorm using a very high-resolution three-dimensional cloud model. **Journal of wind engineering and industrial aerodynamics**, Elsevier, v. 104, p. 547–557, 2012.

ORF, L. G.; ANDERSON, J. R. A numerical study of traveling microbursts. **Monthly Weather Review**, v. 127, n. 6, p. 1244–1258, 1999.

ORF, L. G.; ANDERSON, J. R.; STRAKA, J. M. A three-dimensional numerical analysis of colliding microburst outflow dynamics. **Journal of the atmospheric sciences**, v. 53, n. 17, p. 2490–2511, 1996.

ORF, L. G. et al. Circumferential analysis of a simulated three-dimensional downburst-producing thunderstorm outflow. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 135, p. 182–190, 2014.

PIANIGIANI, G.; POVOLEDO, E.; PÉREZ-PEÑA, R. **Italy Bridge Collapse Leaves 37 Dead**. 2018. <<https://www.nytimes.com/2018/08/14/world/europe/italy-geoa-bridge-collapse.html>>. Accessed: 26 fev. 2020.

PIOMELLI, U.; BALARAS, E. Wall-layer models for large-eddy simulations. **Annual review of fluid mechanics**, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 34, n. 1, p. 349–374, 2002.

PIOMELLI, U. et al. Subgrid-scale backscatter in turbulent and transitional flows. **Physics of Fluids A: Fluid Dynamics**, American Institute of Physics, v. 3, n. 7, p. 1766–1771, 1991.

POPE, S. B. **Turbulent flows**. [S.l.]: IOP Publishing, 2001.

PROCTOR, F.; BOWLES, R. Three-dimensional simulation of the denver 11 July 1988 microburst-producing storm. **Meteorology and Atmospheric Physics**, Springer, v. 49, n. 1-4, p. 107–124, 1992.

PROCTOR, F. H. Numerical simulations of an isolated microburst. part i: Dynamics and structure. **Journal of the atmospheric sciences**, v. 45, n. 21, p. 3137–3160, 1988.

_____. Numerical simulations of an isolated microburst. part ii: Sensitivity experiments. **Journal of the atmospheric sciences**, v. 46, n. 14, p. 2143–2165, 1989.

QU, W.-L. et al. Numerical analysis of factors influencing the downburst wind profiles. In: **The Seventh Asia-Pacific Conference on Wind Engineering, Taipei, Taiwan**. [S.l.: s.n.], 2009.

REN, H.; LAIMA, S.; LI, H. Numerical investigation of very-large-scale motions in a turbulent boundary layer for different roughness. **Energies**, Multidisciplinary Digital Publishing Institute, v. 13, n. 3, p. 659, 2020.

ROBERTS, R. D.; WILSON, J. W. A proposed microburst nowcasting procedure using single-doppler radar. **Journal of Applied Meteorology**, v. 28, n. 4, p. 285–303, 1989.

ROMANIC, D.; PARVU, D.; HANGAN, H. Downburst reconstruction using physical simulation and analytical model with application to urban environments. In: **Proceedings of the First International Conference on Urban Physics (FICUP)**(B. Beckers, T. Pico, S. Jimenez (Eds.)). [S.l.: s.n.], 2016. p. 1–12.

SALINAS, J. S. et al. Turbulent structures in cylindrical density currents in a rotating frame of reference. **Journal of Turbulence**, Taylor & Francis, v. 19, n. 6, p. 463–492, 2018.

SCHUMANN, U. Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli. **Journal of computational physics**, v. 18, p. 376–404, 1975.

SENGUPTA, A.; SARKAR, P. P. Experimental measurement and numerical simulation of an impinging jet with application to thunderstorm microburst winds. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 96, n. 3, p. 345–365, 2008.

SIMPSON, J. E. Gravity currents in the laboratory, atmosphere, and ocean. **Annual Review of Fluid Mechanics**, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 14, n. 1, p. 213–234, 1982.

_____. Gravity currents in the environment. **PROCEEDINGS-INDIAN NATIONAL SCIENCE ACADEMY PART A**, Indian National Science Academy, v. 65, p. 1–26, 1999.

SMAGORINSKY, J. General circulation experiments with the primitive equations: I. the basic experiment. **Monthly weather review**, v. 91, n. 3, p. 99–164, 1963.

SOLARI, G. Emerging issues and new frameworks for wind loading on structures in mixed climates. **Wind Struct**, v. 19, n. 3, p. 295–320, 2014.

SOLARI, G. et al. Characteristics of thunderstorms relevant to the wind loading of structures. **Wind Struct**, v. 20, n. 6, p. 763–791, 2015.

SPIEGEL, E. A.; VERONIS, G. On the boussinesq approximation for a compressible fluid. **The Astrophysical Journal**, v. 131, p. 442, 1960.

SRIVASTAVA, R. A simple model of evaporatively driven downdraft: Application to microburst downdraft. **Journal of the Atmospheric Sciences**, v. 42, n. 10, p. 1004–1023, 1985.

STRAKA, J. M.; ANDERSON, J. R. Numerical simulations of microburst-producing storms: Some results from storms observed during cohmx. **Journal of the atmospheric sciences**, v. 50, n. 10, p. 1329–1348, 1993.

STULL, R. B. **An introduction to boundary layer meteorology**. [S.I.]: Kluwer Academic Publishers, 1988. v. 13.

_____. Static stability—an update. **Bulletin of the American Meteorological Society**, American Meteorological Society, v. 72, n. 10, p. 1521–1530, 1991.

SWITZER, G. F.; PROCTOR, F. H. Terminal area simulation system user's guide-version 10.0. 2014.

VERMEIRE, B. C.; ORF, L. G.; SAVORY, E. Improved modelling of downburst outflows for wind engineering applications using a cooling source approach. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 99, n. 8, p. 801–814, 2011.

_____. A parametric study of downburst line near-surface outflows. **Journal of Wind Engineering and Industrial Aerodynamics**, Elsevier, v. 99, n. 4, p. 226–238, 2011.

VICROY, D. D. A simple, analytical, axisymmetric microburst model for downdraft estimation. 1991.

WAKIMOTO, R. M. Forecasting dry microburst activity over the high plains. **Monthly Weather Review**, v. 113, n. 7, p. 1131–1143, 1985.

WAKIMOTO, R. M.; BRINGI, V. Dual-polarization observations of microbursts associated with intense convection: The 20 July storm during the mist project. **Monthly weather review**, v. 116, n. 8, p. 1521–1539, 1988.

WALLACE, J. M.; HOBBS, P. V. **Atmospheric science: an introductory survey**. [S.I.]: Elsevier, 2006. v. 92.

WILSON, J. W. et al. Microburst wind structure and evaluation of doppler radar for airport wind shear detection. **Journal of Climate and Applied Meteorology**, v. 23, n. 6, p. 898–915, 1984.

WOLFSON, M. M. Characteristics of microbursts in the continental united states. Citeseer, 1988.

WU, J.-Z.; MA, H.-Y.; ZHOU, M.-D. **Vorticity and vortex dynamics**. [S.I.]: Springer Science & Business Media, 2007.

YAO, J.; LUNDGREN, T. Experimental investigation of microbursts. **Experiments in Fluids**, Springer, v. 21, n. 1, p. 17–25, 1996.

ZEYTOUNIAN, R. K. **Asymptotic modeling of atmospheric flows**. [S.I.]: Springer Science & Business Media, 1990.

ZHANG, Y.; HU, H.; SARKAR, P. P. Modeling of microburst outflows using impinging jet and cooling source approaches and their comparison. **Engineering Structures**, Elsevier, v. 56, p. 779–793, 2013.

ZHANG, Y.; SARKAR, P. P.; HU, H. An experimental investigation on the characteristics of fluid–structure interactions of a wind turbine model sited in microburst-like winds. **Journal of Fluids and Structures**, Elsevier, v. 57, p. 206–218, 2015.

Apêndice A – Condições iniciais de contorno

A.1 0/U

```
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: 2.4.0 |
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       binary;
12     class        volVectorField;
13     location     "0";
14     object       U;
15 }
16 // *****
17
18 dimensions      [0 1 -1 0 0 0 0];
19
20 internalField   uniform (0 0 0);
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      pressureInletOutletVelocity;
27         value     uniform (0 0 0);
28     }
29     floor
30     {
31         type      velocityABLWallFunction;
32         print     1;
33         value     uniform (0 0 0);
34     }
35     fixedWalls
36     {
37         type      pressureInletOutletVelocity;
38         value     uniform (0 0 0);
39     }
40     front
41     {
42         type      symmetryPlane;
43     }
44     back
45     {
46         type      symmetryPlane;
47     }
48 }
```

```

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: 2.4.0 |
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        binary;
12     class         volScalarField;
13     location      "0";
14     object        T;
15 }
16 // ***** //
17
18 dimensions      [0 0 0 1 0 0 0];
19
20 internalField    uniform 300;
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      zeroGradient;
27     }
28     floor
29     {
30         type      zeroGradient;
31     }
32     fixedWalls
33     {
34         type      zeroGradient;
35     }
36     front
37     {
38         type      symmetryPlane;
39     }
40     back
41     {
42         type      symmetryPlane;
43     }
44 }

```

```

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: 2.4.0 |
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       binary;
12     class        volScalarField;
13     location     "0";
14     object       p_rgh;
15 }
16 // ***** //
17
18 dimensions      [0 2 -2 0 0 0];
19
20 internalField   uniform 0;
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      totalPressure;
27         rho       rhok;
28         psi       none;
29         gamma     1;
30         p0        uniform 0;
31         value     uniform 0;
32     }
33     floor
34     {
35         type      fixedFluxPressure;
36         gradient  uniform 0;
37         value     uniform 0;
38     }
39     fixedWalls
40     {
41         type      totalPressure;
42         rho       rhok;
43         psi       none;
44         gamma     1;
45         p0        uniform 0;
46         value     uniform 0;
47     }
48     front
49     {
50         type      symmetryPlane;
51     }
52     back
53     {
54         type      symmetryPlane;
55     }
56 }

```

```

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: 2.4.0 |
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        binary;
12     class         volScalarField;
13     location      "0";
14     object        nuSgs;
15 }
16 // ***** //
17
18 dimensions      [0 2 -1 0 0 0];
19
20 internalField    uniform 0;
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      fixedValue;
27         value      uniform 0;
28     }
29     floor
30     {
31         type      fixedValue;
32         value      uniform 0;
33     }
34     fixedWalls
35     {
36         type      fixedValue;
37         value      uniform 0;
38     }
39     front
40     {
41         type      symmetryPlane;
42     }
43     back
44     {
45         type      symmetryPlane;
46     }
47 }

```

```

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: 2.4.0 |
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        binary;
12     class         volScalarField;
13     location      "0";
14     object        kappat;
15 }
16 // ***** //
17
18 dimensions      [0 2 -1 0 0 0];
19
20 internalField    uniform 0;
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      fixedValue;
27         value      uniform 0;
28     }
29     floor
30     {
31         type      fixedValue;
32         value      uniform 0;
33     }
34     fixedWalls
35     {
36         type      fixedValue;
37         value      uniform 0;
38     }
39     front
40     {
41         type      symmetryPlane;
42     }
43     back
44     {
45         type      symmetryPlane;
46     }
47 }

```

```

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: 2.4.0 |
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        binary;
12     class         volSymmTensorField;
13     location      "0";
14     object        Rwall;
15 }
16 // ***** //
17
18 dimensions      [0 2 -2 0 0 0 0];
19
20 internalField    uniform (0 0 0 0 0 0);
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      fixedValue;
27         value      uniform (0 0 0 0 0 0);
28     }
29     floor
30     {
31         type      SchumannGrotzbach;
32         kappa      0.4;
33         z0         uniform 0.01;
34         betaM      16;
35         gammaM     5;
36         averageType planarAverage;
37         value      uniform (0 0 0 0 0 0);
38     }
39     fixedWalls
40     {
41         type      fixedValue;
42         value      uniform (0 0 0 0 0 0);
43     }
44     front
45     {
46         type      symmetryPlane;
47     }
48     back
49     {
50         type      symmetryPlane;
51     }
52 }

```

```

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: 2.4.0 |
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        binary;
12     class         volVectorField;
13     location      "0";
14     object        qwall;
15 }
16 // ***** //
17
18 dimensions      [0 1 -1 1 0 0 0];
19
20 internalField    uniform (0 0 0);
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      fixedValue;
27         value      uniform (0 0 0);
28     }
29     floor
30     {
31         type      fixedValue;
32         value      uniform (0 0 0);
33     }
34     fixedWalls
35     {
36         type      fixedValue;
37         value      uniform (0 0 0);
38     }
39     front
40     {
41         type      symmetryPlane;
42     }
43     back
44     {
45         type      symmetryPlane;
46     }
47 }

```

```

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: 2.4.0 |
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        binary;
12     class         volScalarField;
13     location      "0";
14     object        k;
15 }
16 // ***** //
17
18 dimensions      [0 2 -2 0 0 0];
19
20 internalField    uniform 0.1;
21
22 boundaryField
23 {
24     ceiling
25     {
26         type      zeroGradient;
27     }
28     floor
29     {
30         type      zeroGradient;
31     }
32     fixedWalls
33     {
34         type      zeroGradient;
35     }
36     front
37     {
38         type      symmetryPlane;
39     }
40     back
41     {
42         type      symmetryPlane;
43     }
44 }

```

Apêndice B – Modelo de turbulência e propriedades físicas

B.1 constant/LESProperties

```
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: 1.6 |
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     location     "constant";
14     object       LESProperties;
15 }
16 // ***** //
17 #include        "../setUp"
18
19 LESModel        $LESModel;
20
21 delta          vanDriest;
22
23 SmagorinskyCoeffs
24 {
25     ce          $ce;
26     ck          $ck;
27 }
28
29 SmagorinskyABLCoeffs
30 {
31     ce          $ce;
32     ck          $ck;
33     TName       "T";
34     kappatName  "kappat";
35 }
36
37 oneEqEddyABLCoeffs
38 {
39     ce          $ce;
40     ck          $ck;
41     TName       "T";
42     kappatName  "kappat";
43 }
44
45 printCoeffs    on;
46
47 dynLagrangianCsBoundCoeffs
48 {
49     filter      simple;
50 }
51
52 cubeRootVolCoeffs
53 {
54     deltaCoeff  1;
55 }
```

```
56
57 PrandtlCoeffs
58 {
59     delta          cubeRootVol;
60     cubeRootVolCoeffs
61     {
62         deltaCoeff    1;
63     }
64
65     smoothCoeffs
66     {
67         delta          cubeRootVol;
68         cubeRootVolCoeffs
69         {
70             deltaCoeff    1;
71         }
72
73         maxDeltaRatio  1.1;
74     }
75
76     Cdelta          0.158;
77 }
78
79 vanDriestCoeffs
80 {
81     delta          cubeRootVol;
82     cubeRootVolCoeffs
83     {
84         deltaCoeff    1;
85     }
86
87     smoothCoeffs
88     {
89         delta          cubeRootVol;
90         cubeRootVolCoeffs
91         {
92             deltaCoeff    1;
93         }
94
95         maxDeltaRatio  1.1;
96     }
97
98     Aplus          26;
99     Cdelta          0.158;
100 }
101
102 smoothCoeffs
103 {
104     delta          cubeRootVol;
105     cubeRootVolCoeffs
106     {
107         deltaCoeff    1;
108     }
109
110     maxDeltaRatio  1.1;
111 }
```

```

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: 2.4.0 |
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     location     "constant";
14     object       transportProperties;
15 }
16 // * * * * *
17
18 #include        "../setUp"
19
20 transportModel Newtonian;
21
22 // Molecular viscosity
23 nu             nu [0 2 -1 0 0 0 0] $nu;
24
25 // Reference temperature
26 TRef          TRef [0 0 0 1 0 0 0] $TRef;
27
28 // Prandtl numbers
29 Pr            Pr [0 0 0 0 0 0 0] $Pr;
30 Prt           Prt [0 0 0 0 0 0 0] $Prt;

```

```
1 /*----- C++ -----*/
2 | ===== |
3 | \\ / F ield | OpenFOAM: The Open Source CFD Toolbox |
4 | \\ / O peration | Version: 2.4.x |
5 | \\ / A nd | Web: www.OpenFOAM.org |
6 | \\ M anipulation | |
7 /*-----*/
8
9 // General conditions and parameters.
10 Pr 0.7; // Molecular Prandtl number.
11 Prt 0.33333333; // Turbulent Prandtl number.
12 nu 1.6E-5;
13 TRef 300.0; // Reference potential temperature (K).
14
15
16 // SGS model inputs.
17 LESModel oneEqEddyABL; //KosovicOneEqNBA, SmagorinskyABL // SGS model selection.
18 ce 0.93; // SGS model constant.
19 ck 0.0673; // SGS model constant.
```

```
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: 2.4.0 |
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        uniformDimensionedVectorField;
13     location     "constant";
14     object       g;
15 }
16 // * * * * *
17
18 dimensions      [0 1 -2 0 0 0 0];
19 value           ( 0 0 -9.81);
```

Apêndice C – Soluções numéricas

C.1 system/blockMeshDict

```
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: 2.4.0 |
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 xMax 3500;
18
19 yMaxp 0;
20 yMaxn 3500;
21
22 zmax 4000;
23
24 convertToMeters 1;
25
26 vertices
27 (
28     (0 $yMaxn 0)
29     ($xMax $yMaxn 0)
30     ($xMax $yMaxn $zmax)
31     (0 $yMaxn $zmax)
32
33     (0 $yMaxp 0)
34     ($xMax $yMaxp 0)
35     ($xMax $yMaxp $zmax)
36     (0 $yMaxp $zmax)
37 );
38 blocks
39 (
40     hex (0 1 2 3 4 5 6 7) (350 243 350) simpleGrading (1 71 1)
41 );
42 edges
43 (
44
45 );
46 boundary
47 (
48     ceiling
49     {
50         type patch;
51         faces
52         (
53             (3 7 6 2)
54
55         );
```

```
56     }
57     floor
58     {
59         type wall;
60         faces
61         (
62             (0 1 5 4)
63
64         );
65     }
66     fixedWalls
67     {
68         type patch;
69         faces
70         (
71
72
73             (1 2 6 5)
74             (0 3 2 1)
75         );
76     }
77     front
78     {
79         type symmetryPlane;
80         faces
81         (
82
83             (0 4 7 3)
84
85         );
86     }
87     back
88     {
89         type symmetryPlane;
90         faces
91         (
92
93             (4 5 6 7)
94
95         );
96     }
97 );
98 mergePatchPairs
99 (
100 );
```

```

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: 2.4.x |
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     location     "system";
14     object       controlDict;
15 }
16 // * * * * * //
17 libs ("libSOWFAfiniteVolume.so" "libSOWFAincompressibleLESModels.so");
18
19 application      subCloudPimpleFoam;
20 startFrom        latestTime;
21 startTime        0;
22 stopAt           endTime;
23 endTime         450;
24 deltaT          0.3;
25 writeControl     adjustableRunTime;;
26 writeInterval   2;
27 purgeWrite      0;
28 writeFormat     binary;
29 writePrecision  12;
30 writeCompression off;
31 timeFormat      general;
32 timePrecision   10;
33 runTimeModifiable true;
34 adjustTimeStep true;
35 maxCo          1;
36
37 functions
38 {
39     minMax1
40     {
41         functionObjectLibs ( "libfieldFunctionObjects.so" );
42         type                fieldMinMax;
43         mode                component;
44         fields              (U);
45         outputControl       outputTime;
46     }
47     minMax2
48     {
49         functionObjectLibs ( "libfieldFunctionObjects.so" );
50         type                fieldMinMax;
51         mode                magnitude;
52         fields              (U);
53         outputControl       outputTime;
54     }
55     minMax3
56     {
57         functionObjectLibs ( "libfieldFunctionObjects.so" );
58         type                fieldMinMax;

```

```
59     mode           magnitude;
60     fields         (T);
61     outputControl  outputTime;
62 }
63 fieldAverage1
64 {
65     type           fieldAverage;
66     functionObjectLibs ( "libfieldFunctionObjects.so" );
67     enabled        true;
68     outputControl  outputTime;
69
70     fields
71     (
72         U
73         {
74             mean      on;
75             prime2Mean on;
76             base      time;
77             windowType exact;
78             window    5.0;
79         }
80         p
81         {
82             mean      on;
83             prime2Mean on;
84             base      time;
85             windowType exact;
86             window    5.0;
87         }
88     );
89 }
90 }
```

```

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: 2.4.0 |
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     location     "system";
14     object       fvOptions;
15 }
16 // ***** //
17
18 sourceTemperature
19 {
20     type          scalarCodedSource;
21     active        true;
22     selectionMode all;
23     rampCoeffs   {};
24     scalarCodedSourceCoeffs
25     {
26         fieldNames      (T);
27         redirectType    ramp;
28         codeInclude
29         #{
30
31         #};
32         codeCorrect
33         #{
34             Pout<< "***codeCorrect**" << endl;
35         #};
36         codeAddSup
37         #{
38             const Time& time = mesh().time();
39             const volVectorField& ctrs = mesh_.C();
40             scalarField& QSource = eqn.source();
41             const scalarField& V = mesh_.V();
42
43             scalar  Xo = 0.0;
44             scalar  Yo = 0.0;
45             scalar  Zo = 2000.0;
46             scalar  hx = 1200.0;
47             scalar  hy = 1200.0;
48             scalar  hz = 1800.0;
49             scalar  gt = 0.08;
50             forAll(ctrs, i)
51             {
52
53                 scalar  sqrX = sqr((ctrs[i].component(vector::X) - Xo)/hx);
54                 scalar  sqrY = sqr((ctrs[i].component(vector::Y) - Yo)/hy);
55                 scalar  sqrZ = sqr((ctrs[i].component(vector::Z) - Zo)/hz);
56                 scalar  R = sqrt(sqrX + sqrZ + sqrY);
57
58                 if ((time.value() >= 0) && (time.value() < 120))

```

```

59     {
60     gt = 0.08*sqr(cos(M_PI*((time.value() - 120)/240)));
61     }
62     else if((time.value() >= 120) && (time.value() <= 720))
63     {
64     gt = 0.08*1;
65     }
66     else if((time.value() > 720) && (time.value() <= 840))
67     {
68     gt = 0.08*sqr(cos(M_PI*((time.value() - 720)/240)));
69     }
70     else if(time.value() > 840)
71     {
72     gt = 0.0;
73     }
74     if (R <= 0.5)
75     {
76     QSource[i] = gt*sqr(cos(M_PI*R))*V[i];
77     }
78     else
79     {
80     QSource[i] = 0.0;
81     }
82     }
83     #};
84
85     codeSetValue
86     #{
87     Pout<< "**codeSetValue**" << endl;
88     #};
89
90     // Dummy entry. Make dependent on above to trigger recompilation
91     code
92     #{
93     $codeInclude
94     $codeCorrect
95     $codeAddSup
96     $codeSetValue
97     #};
98     }
99     sourceTimeCoeffs
100     {
101     $scalarCodedSourceCoeffs;
102     }
103 }

```

```

1  /*----- C++ -----*/
2  |=====|
3  | \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
4  | \ \ / O peration | Version: 2.4.0 |
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        fvSchemes;
15 }
16 ddtSchemes
17 {
18     default       backward;
19 }
20 gradSchemes
21 {
22     default       Gauss linear;
23     grad(U)       Gauss linear;
24 }
25 divSchemes
26 {
27     default       none;
28     div(phi,U)    Gauss linear;
29     div(phi,T)    Gauss limitedLinear 1;
30     div(phi,k)    Gauss linear;
31     div(phi,B)    Gauss linear;
32     div(R)        Gauss linear;
33     div(U)        Gauss linear;
34     div(B)        Gauss linear;
35     div(nonlinearStress) Gauss linear;
36     div(Rwall)    Gauss linear;
37     div(qwall)    Gauss linear;
38     div((nuEff*dev(T(grad(U)))) Gauss linear;
39 }
40 laplacianSchemes
41 {
42     default       Gauss linear corrected;
43 }
44 interpolationSchemes
45 {
46     default       linear;
47 }
48 snGradSchemes
49 {
50     default       corrected;
51 }
52 fluxRequired
53 {
54     default       no;
55     p_rgh         ;
56 }

```

C.5 system/fvSolution

```
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: 2.4.0 |
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     location     "system";
14     object       fvSolution;
15 }
16 solvers
17 {
18     p_rgh
19     {
20         solver      PCG;
21         preconditioner DIC;
22         tolerance   1e-8;
23         relTol      0.01;
24         minIter     3;
25     }
26     p_rghFinal
27     {
28         $p_rgh;
29         relTol      0;
30     }
31     "(U|T|k|epsilon|R)"
32     {
33         solver      PBiCG;
34         preconditioner DILU;
35         tolerance   1e-7;
36         relTol      0.1;
37         minIter     3;
38     }
39     "(U|T|k|epsilon|R) Final"
40     {
41         $U;
42         relTol      0.0;
43     }
44 }
45 PIMPLE
46 {
47     momentumPredictor yes;
48     nOuterCorrectors 1;
49     nCorrectors 3;
50     nNonOrthogonalCorrectors 0;
51 }
52 relaxationFactors
53 {
54     fields
55     {
56     }
57     equations
58     {
59         "(U|T|k|epsilon|R)" 1.0;
60         "(U|T|k|epsilon|R) Final" 1.0;
61     }
62 }
```



Apêndice D – Solucionador subCloudPimpleFoam

D.1 subCloudPimpleFoam.C

```
1 /*-----*/
2 ===== |
3 \\ / F ield | OpenFOAM: The Open Source CFD Toolbox
4 \\ / O peration | Website: https://openfoam.org
5 \\ / A nd | Copyright (C) 2011–2018 OpenFOAM Foundation
6 \\ / M anipulation |
7 -----
8 License
9 This file is part of OpenFOAM.
10
11 OpenFOAM is free software: you can redistribute it and/or modify it
12 under the terms of the GNU General Public License as published by
13 the Free Software Foundation, either version 3 of the License, or
14 (at your option) any later version.
15
16 OpenFOAM is distributed in the hope that it will be useful, but WITHOUT
17 ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
18 FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
19 for more details.
20
21 You should have received a copy of the GNU General Public License
22 along with OpenFOAM. If not, see <http://www.gnu.org/licenses/>.
23
24 Application
25 subCloudPimpleFoam
26 -----*/
27 #include "fvCFD.H"
28 #include "singlePhaseTransportModel.H"
29 #include "turbulenceModel.H"
30 #include "pimpleControl.H"
31 #include "radiationModel.H"
32 #include "fvIOptionList.H"
33 #include "wallDist.H"
34 #include "pimpleControl.H"
35 #include "fixedFluxPressureFvPatchScalarField.H"
36 // * * * * * //
37 int main(int argc, char *argv[])
38 {
39     #include "setRootCase.H"
40     #include "createTime.H"
41     #include "createMesh.H"
42     #include "readGravitationalAcceleration.H"
43     #include "createFields.H"
44     #include "createIncompressibleRadiationModel.H"
45     #include "createFvOptions.H"
46     #include "readTimeControls.H"
47     #include "CourantNo.H"
48     #include "setInitialDeltaT.H"
49
50
51
52     pimpleControl pimple(mesh);
53
54     // * * * * * //
55
```

```
56 Info<< "\nStarting time loop\n" << endl;
57
58 while (runTime.loop())
59 {
60     Info<< "Time = " << runTime.timeName() << nl << endl;
61
62     #include "readTimeControls.H"
63     #include "CourantNo.H"
64     #include "setDeltaT.H"
65     // — Pressure-velocity PIMPLE corrector loop
66     while (pimple.loop())
67     {
68         #include "UEqn.H"
69         #include "turbulenceCorrect.H"
70         #include "TEqn.H"
71
72         // — Pressure corrector loop
73         while (pimple.correct())
74         {
75             #include "pEqn.H"
76             #include "turbulenceCorrect.H"
77             #include "TEqn.H"
78         }
79     }
80
81     runTime.write();
82
83     Info<< "ExecutionTime = " << runTime.elapsedCpuTime() << " s"
84         << " ClockTime = " << runTime.elapsedClockTime() << " s"
85         << nl << endl;
86 }
87 Info<< "End\n" << endl;
88 return 0;
89 }
```

```
1 volScalarField T
2 (
3     IOobject
4     (
5         "T",
6         runTime.timeName(),
7         mesh,
8         IOobject::MUST_READ,
9         IOobject::AUTO_WRITE
10    ),
11    mesh
12 );
13 volSymmTensorField Rwall
14 (
15     IOobject
16     (
17         "Rwall",
18         runTime.timeName(),
19         mesh,
20         IOobject::MUST_READ,
21         IOobject::AUTO_WRITE
22    ),
23    mesh
24 );
25 volVectorField qwall
26 (
27     IOobject
28     (
29         "qwall",
30         runTime.timeName(),
31         mesh,
32         IOobject::MUST_READ,
33         IOobject::AUTO_WRITE
34    ),
35    mesh
36 );
37 volScalarField p_rgh
38 (
39     IOobject
40     (
41         "p_rgh",
42         runTime.timeName(),
43         mesh,
44         IOobject::MUST_READ,
45         IOobject::AUTO_WRITE
46    ),
47    mesh
48 );
49 volVectorField U
50 (
51     IOobject
52     (
53         "U",
54         runTime.timeName(),
55         mesh,
56         IOobject::MUST_READ,
57         IOobject::AUTO_WRITE
58    ),
```

```

59     mesh
60 );
61 #include "createPhi.H"
62 #include "readTransportProperties.H"
63 Info<< "Creating turbulence model\n" << endl;
64     autoPtr<incompressible::turbulenceModel> turbulence
65     (
66         incompressible::turbulenceModel::New(U, phi, laminarTransport)
67     );
68 volScalarField rhok
69 (
70     IOobject
71     (
72         "rhok",
73         runTime.timeName(),
74         mesh
75     ),
76     1.0 - (T - TRef)/TRef
77 );
78 volScalarField kappat
79 (
80     IOobject
81     (
82         "kappat",
83         runTime.timeName(),
84         mesh,
85         IOobject::MUST_READ,
86         IOobject::AUTO_WRITE
87     ),
88     mesh
89 );
90 Info<< "Calculating field g.h\n" << endl;
91 volScalarField gh("gh", g & mesh.C());
92 surfaceScalarField ghf("ghf", g & mesh.Cf());
93 volScalarField p
94 (
95     IOobject
96     (
97         "p",
98         runTime.timeName(),
99         mesh,
100        IOobject::NO_READ,
101        IOobject::AUTO_WRITE
102    ),
103    p_rgh
104 );
105 label pRefCell = 0;
106 scalar pRefValue = 0.0;
107     setRefCell
108     (
109         p,
110         p_rgh,
111         mesh.solutionDict().subDict("PIMPLE"),
112         pRefCell,
113         pRefValue
114     );
115 if (p_rgh.needReference())
116 {
117     p += dimensionedScalar

```

```
118 | (
119 |     "p",
120 |     p.dimensions(),
121 |     pRefValue - getRefCellValue(p, pRefCell)
122 | );
123 | }
```

```
1  #include "computeBuoyancyTerm.H"
2  fvVectorMatrix UEqn
3  (
4      fvm::ddt(U)
5      + fvm::div(phi, U)
6      + turbulence->divDevReff(U)
7      + fvc::div(Rwall)
8      ==
9      fvOptions(U)
10 );
11 UEqn.relax();
12 fvOptions.constrain(UEqn);
13 if (pimple.momentumPredictor())
14 {
15     solve
16     (
17         UEqn
18         ==
19         fvc::reconstruct
20         (
21             (
22                 - fvc::snGrad(p_rgh)
23                 + buoyancyTerm
24             ) * mesh.magSf()
25         )
26     );
27     fvOptions.correct(U);
28 }
```

```
1     surfaceScalarField buoyancyTerm = -ghf* fvc::snGrad(rhok);  
2     buoyancyTerm = ((g & mesh.Sf())/mesh.magSf()) * fvc::interpolate(rhok - 1.0);
```

```
1 {
2   volScalarField kappaEff("kappaEff", turbulence->nu()/Pr + kappat);
3   fvScalarMatrix TEqn
4   (
5     fvm::ddt(T)
6     + fvm::div(phi, T)
7     - fvm::laplacian(kappaEff, T)
8     - fvc::div(qwall)
9     ==
10    fvOptions(T)
11   );
12   TEqn.relax();
13   fvOptions.constrain(TEqn);
14   TEqn.solve();
15   fvOptions.correct(T);
16   rhok = 1.0 - (T - TRef)/TRef);
17 }
```

```

1 { volScalarField rAU("rAU", 1.0/UEqn.A());
2   surfaceScalarField rAUf("1|A(U)", fvc::interpolate(rAU));
3   volVectorField HbyA("HbyA", U);
4   HbyA = rAU*UEqn.H();
5   #include "computeBuoyancyTerm.H"
6   surfaceScalarField phig(rAUf * buoyancyTerm * mesh.magSf());
7   surfaceScalarField phiHbyA
8   (
9     "phiHbyA",
10    (fvc::interpolate(HbyA) & mesh.Sf())
11    + rAUf*fvc::ddtCorr(U, phi)
12  );
13  surfaceScalarField phiFixedFlux = phiHbyA;
14  adjustPhi(phiFixedFlux, U, p_rgh);
15  forAll(p_rgh.boundaryField(), patchi)
16  {
17    if (isA<zeroGradientFvPatchScalarField>(p_rgh.boundaryField()[patchi]))
18    {
19      phiHbyA.boundaryField()[patchi] = phiFixedFlux.boundaryField()[patchi];
20    }
21  }
22  phiHbyA += phig;
23  setSnGrad<fixedFluxPressureFvPatchScalarField>
24  (
25    p_rgh.boundaryField(),
26    (
27      phiHbyA.boundaryField()
28      - phiFixedFlux.boundaryField()
29    )/(mesh.magSf()).boundaryField()*rAUf.boundaryField()
30  );
31  // Non-orthogonal corrector loop.
32  while (pimple.correctNonOrthogonal())
33  {
34    fvScalarMatrix p_rghEqn
35    (
36      fvm::laplacian(rAUf, p_rgh) == fvc::div(phiHbyA)
37    );
38    p_rghEqn.setReference(pRefCell, getRefCellValue(p_rgh, pRefCell));
39    p_rghEqn.solve(mesh.solver(p_rgh.select(pimple.finalInnerIter())));
40    if (pimple.finalNonOrthogonalIter())
41    {
42      phi = phiHbyA - p_rghEqn.flux();
43      p_rgh.relax();
44      U = HbyA + rAU*fvc::reconstruct((phig - p_rghEqn.flux())/rAUf);
45      U.correctBoundaryConditions();
46    }
47  }
48  p = p_rgh + gh;
49  if (p_rgh.needReference())
50  {
51    p += dimensionedScalar
52    (
53      "p",
54      p.dimensions(),
55      pRefValue - getRefCellValue(p, pRefCell)
56    );
57    p_rgh = p - gh;
58  }
59 }

```



```
1 {  
2     kappat = turbulence->nut() / Prt;  
3  
4     turbulence->correct();  
5  
6     kappat.correctBoundaryConditions();  
7  
8     qwall.correctBoundaryConditions();  
9  
10    Rwall.correctBoundaryConditions();  
11 }
```

```
1  singlePhaseTransportModel laminarTransport(U, phi);
2
3  dimensionedScalar TRef(laminarTransport.lookup("TRef"));
4
5  dimensionedScalar Pr(laminarTransport.lookup("Pr"));
6
7  dimensionedScalar Prt(laminarTransport.lookup("Prt"));
```