One-dimensional turbulence (ODT): computationally efficient modeling and simulation of turbulent reacting flows

Victoria B. Stephens[a], John C. Hewson[b], David O. Lignell[a,*]

[a] *Department of Chemical Engineering, Brigham Young University, Provo, UT 84602, USA*

[b] *Fire Science and Technology Department, Sandia National Laboratories, Albuquerque, NM, USA*

Abstract

Write this last. About 100 words.

Keywords: turbulence, reacting flows, one-dimensional turbulence

---

Code Metadata

Nr.	Code metadata description	Please fill in this column
C1	Current code version	2.1
C2	Permanent link to code/repository used for this code version	*github.com/BYUignite/ODT*
C3	Code Ocean compute capsule	N/A
C4	Legal Code License	MIT
C5	Code versioning system used	Git
C6	Software code languages, tools, and services used	C++, Python
C7	Compilation requirements, operating environments & dependencies	CMake 3.12+, Cantera, Git, Python 3.x, Doxygen (optional)
C8	If available Link to developer documentation/manual	N/A
C9	Support email for questions	davidlignellbyu.edu

Table 1: Code metadata (mandatory)

*Corresponding author.

Email address: `davidlignell@byu.edu` (David O. Lignell)

1. Motivation and significance

Questions to answer in this section (from SoftwareX template)

1. What's the scientific background and motivation for this software?
2. Why is this important? What problems does it solve?
3. How has the software contributed (and/or how will it contribute in the future) to the process of scientific discovery? Cite papers using the software.
4. In what experimental setting might someone use this software?
5. What related work is there in the literature?
6. What algorithms, other code/software, or ideas are used? Cite them.

Brain dump notes for this section

- TO DO MAYBE: divide into two subsections: (1) motivation/significance and (2) model history and development

TEXT COPIED FROM CYLINDRICAL ODT PAPER

The one-dimensional turbulence (ODT) model was introduced by Kerstein in 1999 [1]. ODT is computationally efficient because it only resolves flows in a single dimension. Turbulent advection is modeled through stochastic eddy events that are implemented by mapping processes on the domain using triplet maps. Eddy events occur concurrently with the solution of unsteady one-dimensional transport equations for momentum and other scalar quantities. A full range of turbulent length scales is modeled. Eddy event locations, sizes, and occurrence rates are specified dynamically and locally using the momentum fields that evolve with the flow; temperature or scalar fields are also used for buoyant flows. Because the model is one-dimensional, it is limited to homogeneous or boundary layer flows such as jets, wakes, mixing layers, and channel flows. These flows, however, are extremely common in turbulence research, and ODT's computational efficiency and resolution of a full range of scales make it a valuable tool that complements experimental and other simulation methods such as direct numerical simulation (DNS). Unlike ODT, DNS resolves all flow structures in three dimensions, but is computationally expensive. ODT is currently formulated for planar simulations, and the model can be run in temporal or spatial modes. In the temporal mode, the unsteady evolution equations evolve on a one-dimensional domain. In the spatial mode, the flow is assumed steady state (though punctuated with eddy events), and one-dimensional ODT line profiles evolve along a streamwise coordinate that replaces the time coordinate, as is done in steady boundary layer flows.

All of these simulated flows used the planar formulation of ODT, even when comparing to experiments of round jets. This comparison is reasonable for jets because the Reynolds number is axially invariant in both constant-property spatial round jets and constant-property temporal planar jets [2]. To compare temporal ODT simulations to spatial experimental simulations, however, time on the ODT line must be converted to the experimental axial spatial location. This is normally done using a mean line velocity (such as the ratio of the integrated momentum flux to the integrated mass flux) [2], but this implies that all fluid parcels on the line have the same axial location time history. This assumption is not ideal for phenomena that are sensitive to time history, such as soot formation, flame extinction and reignition processes, and particle-turbulence interactions.

These limitations in applying the planar ODT formulation to cylindrical flows motivate the work presented here. In this paper, we extend the ODT model from the planar formulation to include cylindrical and spherical formulations. There have been some previous efforts to implement cylindrical and spherical ODT formulations. Krishnamoorthy [**?**] implemented a cylindrical ODT formulation and applied the model in pipe and jet configurations. Lackmann et al. [**?**] implemented a spherical formulation of the linear eddy model (LEM) for engine applications. Here, we give a detailed description of cylindrical and spherical formulations of ODT. While the spherical formulation is included for completeness, we focus on the cylindrical formulation because it is most directly applicable to current and previous ODT research efforts. We defer to the literature for much of the existing ODT model formulation, e.g., [3, 4, 5], and focus on the new cylindrical and spherical geometries. For completeness, however, we include summary information of the ODT model formulation. We present results of the cylindrical formulation applied to pipe flow, a round nonreacting spatial jet, and a round jet flame. A more detailed study of round pipe flow than is possible in this paper is presented by Medina et al. [**?**].

TEXT COPIED FROM ABANDONED JOSS DRAFT

The Stochastic Eddy Cascade (SEC) package is a set of models and tools used to simulate turbulent flow systems in which turbulence is modeled by stochastic processes that map so-called "eddy events" onto the domain. Most notably, SEC includes the most current implementation of the one-dimensional turbulence model (ODT), which has been successfully used to simulate turbulent reacting and nonreacting flows with high accuracy and relatively low computational cost. In addition to ODT, SEC also offers a flamelet model for combustion systems, which is useful for model validation and testing.

Turbulent flows characterize the vast majority of fluid flows in practical engineering applications, and simulations of turbulent flows provide researchers with valuable insights into complex systems, particularly reacting turbulent flows such as combustion processes [?]. Turbulence is a complex phemonenon that occurs over the full range of the flow's length and time scales. As a result, resolving the entire flow field by numerically solving the Navier-Stokes equations of fluid flow, as is done in direct numerical simulations (DNS), requires intense computational resources. DNS is a powerful research tool, but its high computational cost makes this approach intractable for most practical engineering flows. In order to achieve solutions to practical flow problems, turbulence is most often modeled. For example, large-eddy simulation (LES) approaches directly resolve large-scale flow structures but model small-scale motion, which often results in accurate, computationally efficient simulation data. Unfortunately, LES approaches can introduce empiricism into flow simulations, and errors can be difficult to isolate and quantify. The one-dimensional turbulence model (ODT) functionally reverses the LES approach, modeling large-scale turbulent advection (which is relatively well-understood) and directly resolving small-scale flow structures. In previous studies, ODT has been shown capable of attaining accuracy comparable to DNS at a fraction of the computational cost [? ?], making it an attractive model for simulating turbulent flows, particularly in combustion systems.

ODT has been applied to a wide range of flows. Early applications focused on homogenous turbulence, wakes, and mixing layers (Kerstein1999,Kerstein2000,Kerstein2001). Later extension to variable-density flows and a spatial downstream coordinate system facilitated its growth and application to more complex flows, including combustion in jet flames (Echekki2001; Hewson2001; Hewson2002; Lignell2012; Punati2011; Abdelsamie2017; Lignell2017; Goshayeshi2015), counterflow flames (Jozefik2015), wall fires (Monson2016), and sooting flames, (Lignell2015; Hewson2006; Hewson2009; Lignell2015b, Ricks2010), as well as other particle flows (Sun2017; Schmidt2009; Sun2014; Fistler2017). ODT has also served to complement LES through subgrid modeling studies (Cao2008; Schmidt2003; Schmidt2010) and has been applied to various other flow configurations such as double-diffusive interfaces (GonzalezJuez2011), Rayleigh-Taylor mixing (GonzalezJuez2013), and stratified turbulence (Wunsch2001). Most recently, the ODT code was extended to include cylindrical and spherical coordinate systems (Lignell2018; Klein2018; Klein2019). During this implementation, the ODT code was drastically overhauled, resulting in the SEC package presented here. Further detail on the current ODT model formulation and implementation can be found in the literature (Ashurst2005; Kerstein2001; Lignell2013; Lignell2018).

Ongoing research involving ODT and SEC spans multiple research groups and subject areas. ODT is currently being used to investigate soot formation and destruction in ethylene jet flames via parametric modeling studies (Hewson2015; Lansinger2015b). The SEC package framework is also being used to develop the hierarchical parcel swapping model (HiPS), which models turbulence with an economical, hierarchical network that represents the fluid as individual parcels capable of switching positions within the tree (Kerstein2013; Kerstein2014). In the future, SEC may also be extended to include the linear eddy model (LEM) (Kerstein1991) and LES approaches.

2. Software description

2.1. Model description

Describe how ODT works here. Turbulent advection via eddy events implemented by triplet maps. Planar vs. cylindrical configurations. Eddy events happen concurrently with solution of transport equations. Evolution of transport equations in only one dimension. Limited to one-dimensional flows: jets, wakes, mixing layers. This is fine though because those are common in nature and important to research.

TEXT COPIED FROM UNFINISHED CONFERENCE PAPER

This study employs the most current version of the ODT code as in Lignell et. al. [6], and the ODT model in general is described in detail in the literature [6, 3, 1, 5, 4]. The ODT code uses a Lagrangian finite-volume formulation for diffusive advancement, including adaptive mesh refinement [3]. In this approach, combustion dilation causes grid cells to increase or decrease in volume while mass remains constant within each cell. Turbulent advection is modeled with stochastic processes called "eddy events" that map functions on the domain via triplet maps. These eddy events occur concurrently with the solution of unsteady one-dimensional transport equations for mass, momentum, and enthalpy. In sooting flame cases, we also transport species mass fractions and soot particle size distribution properties. Because the ODT model is one-dimensional, it is limited to homogeneous or boundary-layer flows, such as jets, wakes, and mixing layers; these types of flows, however, are common in nature and central to turbulence research.

and whose sizes are drawn randomly from a sample distribution. A given eddy of size l and location x_0 has an eddy timescale t and an associated eddy rate $1 = t$. Eddies occur as a Poisson process in accordance with their given rates. Three ODT parameters control the eddy event process: the eddy rate parameter C scales the rate of occurrence of the eddies; the viscous penalty parameter Z suppresses small eddies; and the large eddy suppression parameter β constrains eddies such that they do not reach over the elapsed simulation time.

2.2. Software Architecture

Things to put in this section (from SoftwareX template)

- overview of overall software architecture
- optional: pictorial overview
- implementation details
  - software dependencies?
  - how to build and run (or leave this in the code README files? maybe brief description here with full details in README.)
  - how to use post processing tools? (TODO: prune post processing tools as much as possible so there's no excess)

Summary of directory structure:

- source: source code lives here; yaml files generated by building yaml in build folder get put here as well
  - more details on directory structure necessary here? not sure if relevant
- build: go here to build the code
  - Run: make yaml (only need to do this once, the first time you download/build the code)
  - set options/paths for cmake in user_config file
  - Run: cmake -C user_config ../source
  - Run: make -j8
- input: input files live here; change input files to specify case details
  - each case type has its own folder; make your own if necessary
  - within each case folder, you should see file input.yaml. this is your all-important input file that makes each case unique
    * talk about most important ODT parameters/options here? or leave to documentation?
- run: go here to actually run the code; don't forget to make your changes to the input file first

  - after building, you'll see odt.x in this folder. you can run it directly (not recommended) or from within one of the bash scripts (better)
  - don't forget to give your case a name; this name will be used to label data files dumped to the data directory

- data: data and runtime files dumped here, organized by case name and individual realizations
  - TO DO: raw/processed file types and what they contain, how to use them effectively

- post: post processing tools located here, organized by case type

- doc: doxygen documentation (if generated) lives here

3. Software Functionalities and Example Cases

Present the major functionalities of the software.

- simulating turbulent flow cases, reacting or nonreacting flows
- can simulate laminar flow cases too
- does these things a whole lot faster than other methods do (LES, DNS especially)
- testing LES subgrid modeling assumptions
- simulating cases that DNS can't get to because needed simulation length is too long (i.e. late-flame phenomena)

Example cases:

- From cylindrical ODT paper:
  1. pipe flow
  2. nonreacting round jet
  3. round jet flame

- Other possible example cases:
  - something to compare with DNS/LES results to illustrate efficiency
  - no sooting cases since no soot in the code

- planar vs. cylindrical comparison

Things to talk about with example cases:

- compare to DNS/LES cases; computational efficiency, accuracy
- how data looks before post processing, how post processing works

Future functionalities here?

- soot stuff
- radiation models
- easier switching between chemistry models?

4. Impact

Questions to answer in this section (from SoftwareX template)

1. How can new research questions be pursued with this software?
   - possibility of parametric studies (much harder with DNS/LES/RANS)
   - study of late-flame soot and radiation interactions, soot emissions as smoke
   - comparative radiation model studies?
2. How does the software improve pursuit of existing research questions?
   - late-flame behavior becomes easier to study
   - validation of LES subgrid models
   - soot stuff, especially late in the flame (because soot moves slowly compared to gas species and therefore short simulation times like in DNS aren't enough to study it effectively)
3. How does the software change the daily practice of its users?
   - cases take hours or days rather than weeks using supercomputer resources
   - test cases can be run on local computers (unlike something like DNS) and as background tasks without disrupting other tasks
   - ODT as a tool complements other approaches, can cover blind spots and be used in validation
4. How widespread is the software? Who uses it? (Within and outside of intended research area and/or group.)

    - BYU group
    - JCH at Sandia
    - Chalmers group in Sweden (Marco Fistler, etc.)
    - German university group (Heiko Schmidt, Juan Media, Marten Klein, etc.)
    - TO DO: find other groups who have used or currently use ODT

5. How is the software used in commercial settings (if any)? Has it led to creation of spin-off companies?
    - No commercial use (I think).

COPIED/PASTED FROM CYLINDRICAL ODT PAPER ODT has been applied by a number of researchers to a wide range of flows. These include homogeneous turbulence [1, 7], wakes [8], mixing layers [1, 5**?** , 4], stratified flows [9], Rayleigh-Benard flows [**?**], buoyant wall flows [**?** **?**], and channel flows [10, 11, 3]. Schmidt et al. [**?**] studied buoyantly-driven cloud-top entrainment with comparisons to water tank experiments. Gonzalez-Juez et al. [**?**] studied reactive Rayleigh-Taylor mixing. A number of ODT simulations of turbulent jet flames have been performed [12, 13, 14, 15**?** , 16]. Other combustion applications include wall fires [17**?**], buoyant pool fires [18, 19, 20], and opposed jet flames [21]. Several studies used DNS to validate ODT for combustion in temporal planar jets [15**?** , 16]. ODT has also been applied as a subgrid model for large eddy simulations (LES) [22, 11, 23] and in particle dynamics simulations in channel flows [10], homogeneous turbulence [7], and jets [2, 24].

Brain dump stuff to include here:

- future functionalities and plans go here instead of last section?

5. Conclusion

Write this part next to last

6. Conflict of Interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

Acknowledgements

This work is supported in part by the National Science Foundation under Grant No. CBET-1403403 and in part by the Sandia National Laboratories Advanced Simulation and Computing Physics and Engineering Models program. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Special thanks to Alan R. Kerstein for his significant contributions to model development. Additional thanks to Heiko Schmidt, Juan Medina, and Marten Klein of Brandenburg University of Technology Cottbus-Senftenberg, Michael Oevermann and Marco Fistler of Chalmers University of Technology, and Vladimir P. Solovjov of Brigham Young University.

References

[1] A. R. Kerstein, One-dimensional turbulence: model formulation and application to homogeneous turbulence, shear flows, and buoyant stratified flows 392 (1999) 277–334. doi:10.1017/S0022112099005376.

[2] G. Sun, J. C. Hewson, D. O. Lignell, Evaluation of stochastic particle dispersion modeling in turbulent round jets 89 (2017) 108–122. doi:10.1016/j.ijmultiphaseflow.2016.10.005.

[3] D. O. Lignell, A. R. Kerstein, G. Sun, E. I. Monson, Mesh adaption for efficient multiscale implementation of one-dimensional turbulence 27 (3-4) (2013) 273–295. doi:10.1007/s00162-012-0267-9.

[4] W. T. Ashurst, A. R. Kerstein, One-dimensional turbulence: Variable-density formulation and application to mixing layers 17 (2). doi:10.1063/1.1847413.

[5] A. R. Kerstein, W. T. Ashurst, S. Wunsch, V. Nilsen, One-dimensional turbulence: vector formulation and application to free shear flows 447 (2001) 85–109. doi:10.1017/S0022112001005778.

[6] D. O. Lignell, V. B. Lansinger, J. Medina, M. Klein, A. R. Kerstein, H. Schmidt, M. Fistler, M. Oevermann, One-dimensional turbulence modeling for cylindrical and spherical flows: model formulation and application 32 (4) (2018) 495–520. doi:10.1007/s00162-018-0465-1.

[7] G. Sun, D. O. Lignell, J. C. Hewson, C. R. Gin, Particle dispersion in homogeneous turbulence using the one-dimensional turbulence model 26 (10) (2014) 103301. doi:10.1063/1.4896555.

[8] A. R. Kerstein, T. D. Dreeben, Prediction of turbulent free shear flow statistics using a simple stochastic model 12 (2) (2000) 418–424. doi:10.1063/1.870319.

[9] S. Wunsch, A. R. Kerstein, A model for layer formation in stably stratified turbulence 13 (3) (2001) 702–712. doi:10.1063/1.1344182.

[10] J. R. Schmidt, J. O. L. Wendt, A. R. Kerstein, Non-equilibrium wall deposition of inertial particles in turbulent flow 137 (2) (2009) 233–257. doi:10.1007/s10955-009-9844-8.

[11] R. C. Schmidt, A. R. Kerstein, S. Wunsch, V. Nilsen, Near-wall les closure based on one-dimensional turbulence modeling 186 (1) (2003) 317–355. doi:10.1016/S0021-9991(03)00071-8.

[12] T. Echekki, A. R. Kerstein, T. D. Dreeben, J.-Y. Chen, 'one-dimensional turbulence' simulation of turbulent jet diffusion flames: model formulation and illustrative applications 125 (3) (2001) 1083–1105. doi:10.1016/S0010-2180(01)00228-0.

[13] J. C. Hewson, A. R. Kerstein, Stochastic simulation of transport and chemical kinetics in turbulent co/h 2 /n 2 flames 5 (4) (2001) 669–697. doi:10.1088/1364-7830/5/4/309.

[14] J. C. Hewson, A. R. Kerstein, Local extinction and reignition in non-premixed turbulent co/h 2 /n 2 jet flames 174 (5-6) (2002) 35–66. doi:10.1080/713713031.

[15] N. Punati, J. C. Sutherland, A. R. Kerstein, E. R. Hawkes, J. H. Chen, An evaluation of the one-dimensional turbulence model: Comparison with direct numerical simulations of co/h2 jets with extinction and reignition 33 (1) (2011) 1515–1522. doi:10.1016/j.proci.2010.06.127.

[16] D. O. Lignell, G. C. Fredline, A. D. Lewis, Comparison of one-dimensional turbulence and direct numerical simulations of soot formation and transport in a nonpremixed ethylene jet flame 35 (2) (2015) 1199–1206. doi:10.1016/j.proci.2014.05.046.

[17] E. I. Monson, D. O. Lignell, M. A. Finney, C. Werner, Z. Jozefik, A. R. Kerstein, R. S. Hintze, Simulation of ethylene wall fires using

the spatially-evolving one-dimensional turbulence model 52 (1) (2016) 167–196. doi:10.1007/s10694-014-0441-2.

[18] A. J. Ricks, J. C. Hewson, A. R. Kerstein, J. P. Gore, S. R. Tieszen, W. T. Ashurst, A spatially developing one-dimensional turbulence (odt) study of soot and enthalpy evolution in meter-scale buoyant turbulent flames 182 (1) (2010) 60–101. doi:10.1080/00102200903297003.

[19] J. C. Hewson, A. J. Ricks, S. R. Tieszen, A. R. Kerstein, R. O. Fox, Conditional-moment closure with differential diffusion for soot evolution in fire, in: Center for Turbulence Research, Proceedings of the Summer Program, Stanford University, 2006.

[20] J. C. Hewson, A. J. Ricks, S. R. Tieszen, A. R. Kerstein, R. O. Fox, On the transport of soot relative to a flame: modeling differential diffusion for soot evolution in fire, in: H. Bockhorn, A. D'Anna, A. F. Sarofim, H. Wang (Eds.), Combustion generated fine carbonaceous particles, KIT Scientific Publishing, 2009, pp. 571–588.

[21] Z. Jozefik, A. R. Kerstein, H. Schmidt, S. Lyra, H. Kolla, J. H. Chen, One-dimensional turbulence modeling of a turbulent counterflow flame with comparison to dns 162 (8) (2015) 2999–3015. doi:10.1016/j.combustflame.2015.05.010.

[22] S. Cao, T. Echekki, A low-dimensional stochastic closure model for combustion large-eddy simulation 9. doi:10.1080/14685240701790714.

[23] R. C. Schmidt, A. R. Kerstein, R. McDermott, Odtles: A multi-scale model for 3d turbulent flow based on one-dimensional turbulence modeling 199 (13-16) (2010) 865–880. doi:10.1016/j.cma.2008.05.028.

[24] B. Goshayeshi, J. C. Sutherland, Prediction of oxy-coal flame stand-off using high-fidelity thermochemical models and the one-dimensional turbulence model 35 (3) (2015) 2829–2837. doi:10.1016/j.proci.2014.07.003.

Current executable software version

Ancillary data table required for sub version of the executable software: (x.1, x.2 etc.) kindly replace examples in right column with the correct information about your executables, and leave the left column as it is.

Nr.	(Executable) software metadata description	Please fill in this column
S1	Current software version	2.1
S2	Permanent link to executables of this version	For example: *https://github.com/combogenomics/DuctApe/releases/tag/DuctApe* − 0.16.4
S3	Legal Software License	MIT license (MIT)
S4	Computing platforms/Operating Systems	Linux, OS X, Microsoft Windows
S5	Installation requirements & dependencies	CMake 3.12+, Cantera, Git, Doxygen (optional)
S6	If available, link to user manual - if formally published include a reference to the publication in the reference list	For example: *http://mozart.github.io/documentation/*
S7	Support email for questions	davidlignell@byu.edu

Table 2: Software metadata (optional)