MFC: User guide

Spencer H. Bryngelson,[1,*] Kevin Schmidmayer,[1] Vedran Coralic,[2] Jomela Meng,[3] Kazuki Maeda,[4] and Tim Colonius[1]

[1] *Division of Engineering and Applied Science, California Institute of Technology, 1200 E California Blvd, Pasadena, CA 91125, USA*
[2] *Amazon*
[3] *Affiliation here*
[4] *University of Washington*

(Dated: June 3, 2019)

I. BOUNDARY CONDITIONS

	#	Description
Normal	−1	Periodic
	−2	Reflective
	−3	Ghost cell extrapolation
	−4	Riemann extrapolation
	−5	Slip wall
Thompson char.	−6	Non-reflecting subsonic buffer
	−7	Non-reflecting subsonic inflow
	−8	Non-reflecting subsonic outflow
	−9	Force-free subsonic outflow
	−10	Constant pressure subsonic outflow
	−11	Supersonic inflow
	−12	Supersonic outflow

TABLE I. Boundary conditions

II. PATCH TYPES

#	Name	Dim.	Smooth	Description and required parameters
1	Line segment	1	N	Requires `x_centroid` and `x_length`.
2	Circle	2	Y	Requires `x[y]_centroid` and `radius`.
3	Rectangle	2	N	Coordinate-aligned. Requires `x[y]_centroid` and `x[y]_length`.
4	Sweep line	2	Y	Not coordinate aligned. Requires `x[y]_centroid` and `normal(i)`.
5	Ellipse	2	Y	Requires `x[y]_centroid` and `radii(i)`.
6	Vortex	2	N	Isentropic flow disturbance. Requires `x[y]_centroid` and `radius`.
7	2D analytical	2	N	Assigns the primitive variables as analytical functions.
8	Sphere	3	Y	Requires `x[y,z]_centroid` and `radius`.
9	Cuboid	3	N	Coordinate-aligned. Requires `x[y,z]_centroid` and `x[y,z]_length`.
10	Cylinder	3	Y	Requires `x[y,z]_centroid`, `radius`, and `x[y,z]_length`.
11	Sweep plane	3	Y	Not coordinate-aligned. Requires `x[y,z]_centroid` and `normal(i)`.
12	Ellipsoid	3	Y	Requires `x[y,z]_centroid` and `radii(i)`.
13	3D analytical	3	N	Assigns the primitive variables as analytical functions.
14	Sph. harmonic	3	N	Generates spherical harmonic perturbations to a sphere.

TABLE II. Patch geometries

[*] spencer@caltech.edu

III. INPUT PARAMETERS

Parameter	Type	Description
case_dir	String	Case script directory
run_time_info	Logical	Output run-time information
nodes	Integer	Number of nodes
ppn	Integer	Number of cores
queue	String	Queue name
walltime	Time	Maximum run time
mail_list	String	Information sent to this email
x[y,z]_domain%beg[end]	Real	Beginning [ending] of the x[y,z]-direction domain
stretch_x[y,z]	Logical	Stretching of the mesh in the x[y,z]-direction
a_x[y,z]	Real	Rate at which the grid is stretched in the x[y,z]-direction
x[y,z]_a	Real	Beginning of the stretching in the negative x[y,z]-direction
x[y,z]_b	Real	Beginning of the stretching in the positive x[y,z]-direction
cyl_coord	Logical	Cylindrical coordinates (2D: Axisymmetric, 3D: Cylindrical)
m	Integer	Number of grid cells in the x-coordinate direction
n	Integer	Number of grid cells in the y-coordinate direction
p	Integer	Number of grid cells in the z-coordinate direction
dt	Real	Time step size
t_step_start	Integer	Simulation starting time step
t_step_stop	Integer	Simulation stopping time step
t_step_save	Integer	How often to output data

TABLE III. Logistics and computational domain parameters

Parameter	Type	Description
num_patches	Integer	Number of initial condition geometric patches
model_eqns	Integer	Multicomponent model: [1] Γ/Π_∞; [2] 5-equation; [3] 6-equation
alt_soundspeed	Logical	Alternate sound speed and $K\nabla\cdot\boldsymbol{u}$ for 5-equation model
num_fluids	Integer	Number of fluids/components present in the flow
adv_alphan	Logical	Equations for all N volume fractions (instead of $N-1$)
mpp_lim	Logical	Mixture physical parameters limits
mixture_err	Logical	Mixture properties correction
time_stepper	Integer	Runge–Kutta order [1–5]
weno_vars	Integer	WENO reconstruction on [1] Conservative; [2] Primitive variables
weno_order	Integer	WENO order [1,3,5]
weno_eps	Real	WENO perturbation (avoid division by zero)
char_decomp	Logical	Characteristic decomposition
avg_state	Integer	Averaged state evaluation method: [1] Roe average; [2] Arithmetic mean
mapped_weno	Logical	WENO with mapping of nonlinear weights
null_weights	Logical	Null undesired WENO weights
mp_weno	Logical	Monotonicity preserving WENO
riemann_solver	Integer	Riemann solver algorithm: [1] HLL; [2] HLLC; [3] Exact
wave_speeds	Integer	Wave-speed estimation: [1] Direct (Batten et al. 1997); [2] Pressure-velocity (Toro 1999)
commute_err	Logical	Commutative error correction via cell-interior quadrature
split_err	Logical	Dimensional splitting error correction via cell-boundary
bc_x[y,z]%beg[end]	Integer	Beginning [ending] boundary condition in the x[y,z]-direction (negative integer, see table I)

TABLE IV. Simulation algorithm parameters

Parameter	Type	Description
`format`	Integer	Output format. [1]: Silo-HDF5; [2] Binary
`precision`	Integer	[1] Single; [2] Double
`parallel_io`	Logical	Parallel I/O
`cons_vars_wrt`	Logical	Write conservative variables
`prim_vars_wrt`	Logical	Write primitive variables
`fourier_decomp`	Logical	Apply a Fourier decomposition to the output variables
`alpha_rho_wrt(i)`	Logical	Add the partial density of the fluid `i` to the database
`rho_wrt`	Logical	Add the mixture density to the database
`mom_wrt(i)`	Logical	Add the `i`-direction momentum to the database
`vel_wrt(i)`	Logical	Add the `i`-direction velocity to the database
`E_wrt`	Logical	Add the total energy to the database
`pres_wrt`	Logical	Add the pressure to the database
`alpha_wrt(i)`	Logical	Add the volume fraction of fluid `i` to the database
`gamma_wrt`	Logical	Add the specific heat ratio function to the database
`heat_ratio_wrt`	Logical	Add the specific heat ratio to the database
`pi_inf_wrt`	Logical	Add the liquid stiffness function to the database
`pres_inf_wrt`	Logical	Add the liquid stiffness to the formatted database
`c_wrt`	Logical	Add the sound speed to the database
`omega_wrt(i)`	Logical	Add the `i`-direction vorticity to the database
`schlieren_wrt`	Logical	Write numerical schlieren
`fd_order`	Integer	Order [1,2,4] finite differences for the numerical Schlieren function
`schlieren_alpha(i)`	Real	Numerical Schlieren computed via `alpha(i)`
`probe_wrt`	Logical	Write the flow chosen probes data files for each time step
`num_probes`	Integer	Number of probes
`probe(i)%x[y,z]`	Real	Coordinates of probe `i`
`com_wrt(i)`	Logical	Add the center of mass of fluid `i` to the database
`cb_wrt(i)`	Logical	Add coherent body data of fluid `i` to the database

TABLE V. Formatted database output parameters

Parameter	Type	Description
`alter_patch(i)`	Logical	Alter the `i`-th patch
`geometry`	Integer	Geometry configuration of the patch (see table II)
`x[y,z]_centroid`	Real	Centroid of the applied geometry in the x[y,z]-direction
`length_x[y,z]`	Real	Length, if applicable, in the x[y,z]-direction
`radius`	Real	Radius, if applicable, of the applied geometry
`smoothen`	Logical	Smoothen the applied patch
`smooth_patch_id`	Integer	Smoothen of the applied patch with another patch
`smooth_coeff`	Real	Smoothen coefficient
`alpha(i)`	Real	Volume fraction of fluid `i`
`alpha_rho(i)`	Real	Partial density of fluid `i`
`pres`	Real	Pressure
`vel(i)`	Real	Velocity in direction `i`

TABLE VI. Patch parameters. All parameters should be prepended with `patch_icpp(j)%` where `j` is the patch index.

Parameter	Type	Description
`gamma`	Real	Stiffened-gas parameter Γ of fluid `i`: Specific heat ratio
`pi_inf`	Real	Stiffened-gas parameter Π_∞ of fluid `i`: Liquid stiffness
Properties used only for non-polytropic bubble compression model		
`mu_l0`	Real	Liquid viscosity (only specify in liquid phase)
`ss`	Real	Surface tension (only specify in liquid phase)
`pv`	Real	Vapor pressure (only specify in liquid phase)
`gamma_v[n]`	Real	Water [air] compression model property (see Ando 2010)
`M_v[n]`	Real	Water [air] compression model property (see Ando 2010)
`mu_v[n]`	Real	Water [air] compression model property (see Ando 2010)
`k_v[n]`	Real	Water [air] compression model property (see Ando 2010)

TABLE VII. Fluid properties. All parameters should be prepended with `fluid_pp(i)%` where `i` is the fluid index.

Parameter	Type	Description
`Monopole`	Logical	Acoustic source terms
`num_mono`	Integer	Number of acoustic sources
		Properties of acoustic source `i`
`Mono(i)%pulse`	Integer	Type of pulse. [1] Sine [2] Gaussian [3] Square
`Mono(i)%npulse`	Integer	Number of pulse cycles
`Mono(i)%support`	Integer	Spatial support [1] Delta function [2] Finite width (2D) [3] Support for finite line/patch
`Mono(i)%loc(j)`	Real	Location of source in coordinate direction `j`
`Mono(i)%dir`	Real	Direction of propagation
`Mono(i)%mag`	Real	Pulse magnitude
`Mono(i)%length`	Real	Spatial pulse length

TABLE VIII. Acoustic source terms.

Parameter	Type	Description
`bubbles`	Logical	Ensemble-averaged bubble modeling
`bubble_model`	Integer	[1] Gilmore; [2] Keller–Miksis
`polytropic`	Logical	Polytropic gas compression
`thermal`	Integer	Thermal model: [1] Adiabatic; [2] Isothermal; [3] Transfer
`R0ref`	Real	Reference bubble radius
`nb`	Integer	Number of bins: [1] Monodisperse; [> 1] Polydisperse
`Ca`	Real	Cavitation number
`Web`	Real	Weber number
`Re_inv`	Real	Inverse Reynolds number

TABLE IX. Ensemble-averaged bubble model parameters.

Parameter	Type	Description
`weno_avg`	Logical	Averaged left/right cell-boundary states (for Re and We)
`weno_Re_flux`	Logical	WENO reconstruct velocity gradients for viscous stress tensor
`regularization`	Logical	Regularization algorithm of Tiwari et al. (2013)
`reg_eps`	Real	Interface thickness parameter for regularization terms
`tvd_riemann_flux`	Logical	Apply TVD flux limiter to cell edges inside Riemann solver
`tvd_rhs_flux`	Logical	Apply TVD flux limiter to intercell fluxes outside Riemann solver
`tvd_wave_speeds`	Logical	TVD wave-speeds for flux computation inside Riemann solver
`flux_lim`	Integer	Choice of flux limiter: [1] Minmod; [2] MC; [3] Ospre; [4] Superbee; [5] Sweby; [6] van Albada; [7] van Leer.
`We_riemann_flux`	Logical	Capillary effects in the Riemann solver
`We_rhs_flux`	Logical	Capillary effects using a conservative formulation
`We_src`	Logical	Capillary effects using a non-conservative formulation
`We_wave_speeds`	Logical	Capillary effects when computing the contact wave speed
`lsq_deriv`	Logical	Use linear least squares to calculate normals and curvatures
`alt_crv`	Logical	Alternate curvature definition

TABLE X. Experimental features

IV. NAMING CONVENTIONS

Variable	Description
*_sf	Scalar field
*_vf	Vector field
*_pp	Physical parameters
*[K,L,R]	WENO-reconstructed cell averages
*_avg	Roe/arithmetic average
*_cb	Cell boundary
*_cc	Cell center
*_cbc	Characteristic boundary conditions
*_cons_*	Conservative
*_prim_*	Primitive
gm_*	Gradient magnitude
*_ndqp	Normal direction Gaussian quadrature points
*_qp	Cell-interior Gaussian quadrature points
un_*	Unit-normal
dgm_*	Curvature (derived gradient magnitude)
*_icpp	Initial condition patch parameters
*_idx	Indices of first and last (object)
cont_*	Continuity equations
mom_*	Momentum equations
E_*	Total energy equation
adv_*	Volume fraction equations
*_id	Identifier
dflt_*	Default value
*_fp	???
orig_*	Original variable
q_*	Cell-average conservative or primitive variables
q[L,R]_*	Left[right] WENO-reconstructed cell-boundary values
dq_*	First-order spatial derivatives
*_rs	Riemann solver variables
*_src	Source terms
*_gsrc	Geometric source terms
[lo,hi]_*	Related to TVD options
*_IC	Inter-cell
*_ts	Time-stage (for time-stepper algorithm)
wa_*	WENO average
crv_*	Geometrical curvature of the material interfaces

TABLE XI. Code variables