Modeling turbulent flow over fractal trees with renormalized numerical simulation

Stuart Chester [a], Charles Meneveau [a,*], Marc B. Parlange [a,b]

[a] *Center for Environmental and Applied Fluid Mechanics, Johns Hopkins University, Latrobe Hall, 3400 N. Charles Street, Baltimore, MD 21218, USA*
[b] *School of Architecture, Civil and Environmental Engineering, Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*

Abstract

High-Reynolds number flow over tree-like fractals is considered, with emphasis on the drag forces produced. Fractal objects display large scale-disparity and complexity while being amenable to a simple and standardized description. Hence, they offer an elegant idealization of the actual boundaries in practical applications where turbulence interacts with boundaries that are characterized by multiple length-scales. First, using large-eddy-simulation of flow over prefractal shapes with increasing numbers of branch generations, the dependence of the tree drag on the inner cutoff-scale of the fractal is studied. It is found that the convergence of the drag coefficient towards a value that is independent of inner cutoff-scale is very slow. In order to address this fundamental difficulty and avoid the need to resolve all the small-scale branches of the fractal, a new numerical modeling technique called renormalized numerical simulation (RNS) is introduced. RNS models the drag of the unresolved branches using drag coefficients measured from both resolved branches and unresolved branches as modeled in previous iterations of the procedure. The RNS technique and its convergence properties are tested by means of a series of simulations using different levels of resolution. Then, RNS is used to investigate the influence of the tree fractal dimension on the drag coefficient. The increase of the drag with fractal dimension is quantified for two types of tree geometry, in two flow configurations. Results illustrate that RNS enables numerical modeling of physical processes associated with fractal geometries using affordable computational resolution.
© 2006 Elsevier Inc. All rights reserved.

Keywords: Fractals; Large-eddy simulation; Renormalized numerical simulation

1. Introduction

A fundamental challenge in studying flow in many natural systems, e.g., tree canopies [1], coral reefs [2,3], respiratory branching networks [4–6], and soils [7], is the treatment of the wide range of length-scales involved

---

* Corresponding author. Tel.: +1 410 516 7802; fax: +1 410 516 7254.
  E-mail address: meneveau@jhu.edu (C. Meneveau).

in the fluid–boundary interactions. Many such physical systems with large-scale disparities, e.g., lungs, exhibit fractal patterns [8] and therefore have scale-invariant properties. Thus, the study of transport across fractal boundaries is a natural starting point in the study of transport across multi-scale boundaries in general. However, little is known about the transport properties, e.g., drag forces, associated with fractal boundaries in high-Reynolds number (Re) flows.

A number of previous studies have considered harmonic fields in the vicinity of fractal boundaries [9,10] and the associated diffusive transport rates [11,12]. In momentum transport, the low-Re hydrodynamic force on a fractal surface composed of small amplitude sinusoids is known [13] to exhibit a broad multifractal spectrum. The scaling of the drag force and rate of dissipation, and whether these quantities diverge, has also been investigated for certain classes of fractal surfaces in low-Re, low-amplitude oscillatory flows [14–16]. The low-Re hydrodynamic properties of fractal aggregates have also been investigated [17,18]. At high-Re, fractal objects inject energy into a flow over a range of scales, and the influence of this on their wake turbulence has been studied experimentally [19,20]. There is also a large body of literature on modeling vegetation canopy flow [21–23]. Although previous computational studies have not focused on high-Re flow over fractal boundaries, there are numerous techniques for dealing with effects of small-scale surface geometry, i.e., roughness that is separated in scale from the "main" geometry of the flow. In high-Re simulations, the drag effect of such unresolved geometric features (as well as canopy flows) can be taken into account by adding a momentum sink (per unit volume) of the form

$$\boldsymbol{f}(\boldsymbol{x}) = -\frac{1}{2}\rho c_{\mathrm{D}} A(\boldsymbol{x}) |\tilde{\boldsymbol{u}}(\boldsymbol{x})| \tilde{\boldsymbol{u}}(\boldsymbol{x}), \tag{1}$$

to the momentum equations. Here, c_{D} is a drag coefficient, and A is a representative area (per unit volume). This form of momentum sink has been used in a variety of studies where the roughness elements are completely unresolved [24–26]. A drag term representing the effect of unresolved terrain is investigated in [27]. In [28], an analogy with Large Eddy Simulations (LES) is made as the roughness is partially resolved. The resolved part is represented using the immersed boundary method of [29], while the unresolved part is represented using a term similar to Eq. (1). The drag coefficient $c_{\mathrm{D}} = 0.2$ is prescribed, mainly to avoid numerical stability issues they found to be associated with $c_{\mathrm{D}} = 0.5$ used by [25]. In each of the above studies, c_{D} (or the product $c_{\mathrm{D}}A$) was either tuned specifically for the flow under investigation, or it was prescribed in an ad hoc fashion (e.g., to avoid numerical stability issues). This is analogous to the way in which the model coefficient is prescribed in the classical Smagorinsky SGS model in LES, and similar limitations exist on the applicability of a given c_{D} to different flow configurations and flow regimes. In light of this, a technique that could be used to determine c_{D} in a less ad hoc fashion is desirable.

A number of techniques have been developed to treat multi-scale problems with scale-invariant properties. The approach used in renormalization group theory [30–32] to tackle problems with a wide range of scales is to solve a simplified version of the problem at small scales, and then calculate the aggregate effect of the small scales on the large scales. In the remaining large-scale problem, the effects of the small scales appear as effective (renormalized) coefficients. This process of learning from solvable small-scale problems to predict large-scale effective properties is termed "up-scaling". On the other hand, "down-scaling" approaches, where large-scales are used to model small-scale properties, have also been effective in areas such as large-eddy simulation (LES) of turbulent flows. In the dynamic subgrid-scale (SGS) model, resolved-scale information is used to help determine SGS eddy diffusion coefficients [33–35]. Resolved-scale velocity fields have also been used to develop SGS models based on synthetic velocity fields generated using fractal interpolation [36,37].

With the increases in available computing capacity, it is now possible to consider numerical investigations of flow over fractal surfaces. Modeling techniques designed to take advantage of scale-invariance will aid in simulations of these computationally expensive problems for applications, e.g., environmental flows. Here, we numerically study the drag force exerted by a three-dimensional, very high-Re number, turbulent flow on idealized fractal trees. To simplify, the direct effect of molecular viscosity is neglected. The detailed flow configuration envisioned in this study is described in Section 2. Then, in Section 3, we present a series of LES where an increasing number of branches of the tree are explicitly resolved on the computational mesh. The drag coefficient is obtained as a function of the number of branch generations. The results of this study reveal that the convergence towards an asymptotic drag coefficient is extremely slow and that achieving the asymptotic

value would require an impractically large number of branch generations to be resolved in a simulation. Thus, a new method, called renormalized numerical simulation (RNS), is developed in Section 4. RNS takes advantage of scale-invariance in the problem to model the forces on the fluid due to the smallest branches, which cannot be resolved, in a recursive manner. The technique is tested and then applied to predict the drag coefficient as a function of tree fractal dimension and type of fractal tree. Discussion and conclusions are presented in Section 5.

2. Flow configuration

The flow considered in this study is turbulent flow over fractal trees. For the later convenience of using spanwise periodic boundary conditions, the trees are arranged as a single spanwise row placed on a horizontal surface, perpendicular to the flow. The case of many rows (i.e., a lattice) of trees is also considered later, in Section 4.4. For simplicity, we consider trees where the fractal construction is confined to a plane perpendicular to the main flow direction, and we assume the trunk and branches have square cross-sections. The trees are assumed to lie entirely within a turbulent boundary layer developed by the shear from the horizontal surface on which the trees stand. The focus of the present paper is not on turbulence structure created in an actual vegetation canopy. Instead, the focus is on development of techniques to compute drag forces on multi-scale objects. The idealized geometry we consider offers computational and analytical simplifications while preserving essential features of multi-scale boundaries, as a stepping-stone towards simulation of realistic canopy flow.

2.1. Tree geometry

A schematic of the highly idealized tree geometry is shown in Fig. 1. The mean flow is assumed to be along the *x*-direction, and the trees are spaced a distance L apart from each other in the (spanwise) *y*-direction. The plane $z = 0$ defines the horizontal surface, or ground, upon which the trees stand. The overall aim of this study is to predict the drag forces that the trees apply to the fluid flowing over them. It is assumed that the flow Re is so large that the direct effects of viscosity may be neglected in calculation of the drag forces on the trees. In reality, at any fixed Re, there will always be some smallest-scale branch generation at which the local Re becomes small and viscosity begins to play an important role. In this initial study, attention is restricted to the limiting case where even the flow over the very small-scale branches is dominated by inertia effects.

Each tree has multiple generations of branches, with generation 0 being the trunk. To each branch at each generation, N_B smaller branches representing the next generation are attached. To simplify, the branches are square cylinders and all the branches at a given generation have the same size. It is assumed that the trees are self-similar, so that a constant scale ratio $r < 1$ exists between successive branch generations, i.e., the length l_g
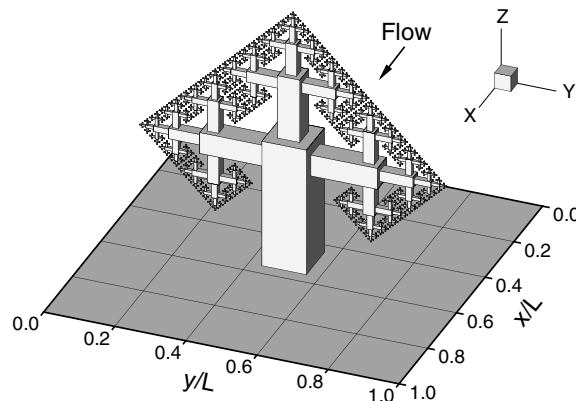

Fig. 1. Schematic of fractal tree geometry considered. The number of sub-branches per branch is $N_B = 3$, and the scale ratio between successive branch generations is $r = 1/2$.

and diameter d_g of a branch at generation $g \geqslant 0$ are related to those at generation $g + 1$ by $l_{g+1} = rl_g$ and $d_{g+1} = rd_g$. The tree geometry can be described by specifying the trunk shape and an iterated function system [38] (IFS) $\{w_i(x): i = 0, \ldots, N_B - 1\}$, with each w_i describing how to map the trunk T to its ith sub-branch. A branch b at generation g can be represented as $b = w_{b_g} \circ \cdots \circ w_{b_1}(T)$, with each b_j in the range $0, \ldots, N_B - 1$. It is assumed that each function in the IFS is invertible, and that the branches are non-overlapping.

In the geometry shown in Fig. 1, which is the test case in Sections 3 and 4.3, each branch has $N_B = 3$ sub-branches and the scale ratio is $r = 1/2$. The similarity fractal dimension [8] is $D = \log N_B / \log r^{-1} \approx 1.58$. The tree trunk is a square cylinder with diameter (square-edge) $d_0 = L/8$ and height $l_0 = 3L/8$. An IFS that describes this tree geometry, relative to coordinates with origin at the base of the trunk, x-axis aligned with the mean flow, and z-axis aligned with the (vertical) trunk axis, is

$$w_i(x) = rR_i x + s_i. \tag{2}$$

In this expression, $R_0 = R_x(\pi/2)$, $R_1 = R_x(0)$, $R_2 = R_x(-\pi/2)$, $R_x(\theta)$ denotes a rotation matrix that rotates through angle θ about the x-axis (in a right-handed sense), $s_0 = (0, -d_0/2, l_0 - d_0/2)$, $s_1 = (0, 0, l_0)$, and $s_2 = (0, d_0/2, l_0 - d_0/2)$.

2.2. A priori drag estimate

As a global measure of the total tree drag, the time-averaged effective total drag coefficient C_T for the entire domain is used. It is defined as $C_T = \overline{F}_T / (\frac{1}{2} \rho U^2 A)$, where $\overline{F}_T$ is the time-averaged total x-direction drag force the fluid applies on the tree, U is the mean streamwise velocity in the domain, and $A = L^2$ is the total frontal area of the domain.

A first, highly approximate, estimate of the drag coefficient $C_T^{(\text{ind})}(g)$ of the tree with g generations of branches is obtained by treating the tree as a superposition of independent square cylinders, each in a uniform flow U, and neglecting end-effects as well as interactions among branches of the same and different generations (e.g., screening). Summing the resulting finite geometric series leads to the estimate

$$C_T^{(\text{ind})}(g) = C_{\text{sq}} \frac{l_0 d_0}{L^2} \frac{1 - (N_B r^2)^{g+1}}{1 - N_B r^2}, \tag{3}$$

where $C_{\text{sq}} \approx 2.1$ is the usual square-cylinder drag coefficient [39], and it is assumed that $N_B r^2 < 1$ (i.e., $D < 2$) so that the expression converges as $g \to \infty$. For the tree in Fig. 1, this estimate yields $C_T^{(\text{ind})}(g \to \infty) \approx 0.39$. The applicability of Eq. (3) to the current flow is limited by the presence of the lower wall (causing the incoming log-profile), the dependence of the flow field felt by a given branch on the presence of the other branches, and the inflow turbulence (which may alter the effective C_{sq}). In the following section, $C_T(g)$ is obtained numerically using LES for different generations $g = 0, 1, 2$ and 3.

3. Branch-resolved simulation

In this section, turbulent flow over finite-generation (i.e., prefractal) approximations to the trees described in Section 2 is simulated. These simulations use only tree geometries that can be directly represented, i.e., resolved, and the effects of unresolved branches are entirely neglected. These simulations are referred to as branch-resolved simulations (BRS). However, the flow itself is turbulent and all the scales of motion cannot be resolved, so the LES technique is used to account for unresolved motion of the fluid. The purpose is to demonstrate the computational difficulty of this multi-scale flow problem and the need to account for unresolved geometry using a new model.

3.1. Simulation methods

The LES technique is used to simulate the high-Re turbulence in the flow surrounding the solid trees. All simulations use the Smagorinsky SGS model [40,41] (the coefficient used is $c_S = 0.16$), with a standard wall-damping function to decrease the coefficient near walls and tree surfaces (the damping function of Ref. [42] is used, with $n = 2$). More advanced SGS closures, such as scale-dependent dynamic models [43,44], have also

been used in some of our simulations. Overall, the results are similar, but not exactly the same. In order to focus this study on the drag forces generated by the fractal boundary and not on the subgrid modeling of the turbulence in the bulk flow, all results presented in this paper use the traditional non-dynamic Smagorinsky model with wall damping, as described above. The molecular viscosity is neglected, consistent with the assumption of very high-Re flow.

The incompressible LES equations are solved using a numerical algorithm that has been described in previous studies [45]. It uses pseudospectral discretization for horizontal (x,y)-planes and second-order centered finite difference discretization in the vertical z-direction. The application of the LES grid filter (denoted by a tilde) is implicit, i.e., no explicit grid filtering operation is performed during the simulations. The immersed boundary method [46–49] is used to add effects of the resolved parts of the trees by forcing the velocity inside them to zero. The second-order Adams–Bashforth time advancement can be written in two stages as

$$\tilde{\boldsymbol{u}}^* = \tilde{\boldsymbol{u}}^n + (3\Delta t/2)[(\tilde{\boldsymbol{u}} \times \tilde{\boldsymbol{\omega}})^n - \nabla \cdot \boldsymbol{\tau}^n] - (\Delta t/2)[(\tilde{\boldsymbol{u}} \times \tilde{\boldsymbol{\omega}})^{n-1} - \nabla \cdot \boldsymbol{\tau}^{n-1} - \nabla\phi^{n-1}], \tag{4}$$

$$\tilde{\boldsymbol{u}}^{n+1} = \tilde{\boldsymbol{u}}^* + \Delta t(\boldsymbol{r}^n/\rho - 3\nabla\phi^n/2), \tag{5}$$

where $\tilde{\boldsymbol{u}}$ is the filtered velocity, $\tilde{\boldsymbol{u}}^*$ is an intermediate velocity, $\tilde{\boldsymbol{\omega}}$ is the filtered vorticity, $\boldsymbol{\tau}$ is the SGS stress tensor, and $\phi = (\tilde{p}/\rho + |\tilde{\boldsymbol{u}}|^2/2)$ is related to the filtered pressure $\tilde{p}$, ρ is the fluid density, $\boldsymbol{r}$ is the immersed boundary force (defined per unit volume, for later convenience), and Δt is the time increment. The superscripts n and $n + 1$ denote the time step associated with each quantity. The variable ϕ^n is determined by solving the Poisson equation obtained by taking the divergence of Eq. (5) and assuming $\nabla \cdot \tilde{\boldsymbol{u}}^{n+1} = \nabla \cdot \boldsymbol{r}^n = 0$.

A signed distance function φ is used to keep track of the points inside the resolved parts of the trees ($\varphi \leqslant 0$) where the immersed boundary forcing will be applied, and the fluid points ($\varphi > 0$). The immersed boundary force only acts on points inside or on the surface of resolved branches and drives the fluid velocity to zero at these points:

$$\boldsymbol{r}^n(\boldsymbol{x}) = \begin{cases} \rho(3\nabla\phi^n/2 - \tilde{\boldsymbol{u}}^*/\Delta t) : & \varphi(\boldsymbol{x}) \leqslant 0 \\ \boldsymbol{0} : & \varphi(\boldsymbol{x}) > 0 \end{cases}. \tag{6}$$

Since the LES presented in this study are not wall-resolving, a wall model is required to represent the drag that the resolved tree surface applies to the fluid flowing past it. There exists presently no accurate wall model applicable for general geometries. Therefore, we will follow the default standard approach [50] to simply use a wall model developed for flat walls [42]. In following this approach, we apply a stress at the resolved tree surfaces consistent with a log-law velocity profile near these surfaces (described in detail below). It is clear that in flows where the skin friction is important, improvements can be made by using more complex wall models. In our simulations, however, the skin-friction-like effects caused by wall roughness z_0 are small compared to the form drag, and so the error incurred by using our simple wall model is not large. In fact, these errors are shown to be small by the favorable comparison with experiment for the square cylinder flow presented in Appendix A. The approach is to modify the SGS stress $\boldsymbol{\tau}$ at points inside and in the vicinity of IB. This is done in three steps, which are illustrated in Fig. 2.

Step 1. Specify the stress at any point $\boldsymbol{x}$ in the band $0 \leqslant \varphi(\boldsymbol{x}) \leqslant \delta$, i.e., any point outside but within distance δ of the IB, including points on the IB itself. Let $\hat{\boldsymbol{n}} = \nabla\varphi$ denote an approximate surface normal passing through $\boldsymbol{x}$ (note $|\nabla\varphi| = 1$ for a signed distance function). Let $\boldsymbol{v}$ denote the fluid velocity at a distance δ from the immersed boundary along the line passing through $\boldsymbol{x}$ and parallel to $\hat{\boldsymbol{n}}$. We use $\delta/h = 1.1$, where h is the mesh spacing (assumed equal in all three Cartesian directions). Then $\boldsymbol{v}$ and $\hat{\boldsymbol{n}}$ define a local coordinate system with origin at $\boldsymbol{x}$ and directions $\boldsymbol{e}_1' = \hat{\boldsymbol{v}}_t$, $\boldsymbol{e}_2' = \hat{\boldsymbol{n}} \times \hat{\boldsymbol{v}}_t$, and $\boldsymbol{e}_3' = \hat{\boldsymbol{n}}$, where $\boldsymbol{v}_t = \boldsymbol{v} - (\boldsymbol{v} \cdot \hat{\boldsymbol{n}})\hat{\boldsymbol{n}}$ is the tangential part of $\boldsymbol{v}$. In these coordinates, the tangential velocity magnitude is assumed to have an instantaneous, rough-wall log-law profile at distances from the wall less than δ, so the local wall shear stress is related to $\boldsymbol{v}$ by

$$\tau_{\mathrm{w}} = -\rho[\kappa|\boldsymbol{v}_t|/\ln(1 + \delta/z_0)]^2, \tag{7}$$

where z_0 is the roughness length of the IB surface, $\kappa = 0.4$ is the von Kármán constant, and the minus sign is for consistency with the SGS stress sign convention. The tree surface roughness length z_0 represents a skin-friction-like drag on the flow caused by the small-scale surface roughness of the tree (e.g., bark). The primary motivation for introducing a small-scale surface roughness z_0 is that the log-profile used in our wall model

Fig. 2. Schematic showing the three steps in the immersed boundary treatment. Here, $x^{(1)}$ denotes a point in the band of thickness δ (=1.1h) in which τ_w is used to calculate the shear stress components $\tau'_{13} = \tau'_{31}$ in Step 1. The point $x^{(2)}$ denotes a point where the stress is obtained by extrapolation during Step 2. Finally, $x^{(3)}$ is a point where the stress is calculated by performing a Laplacian smoothing in the pseudospectral (x, y)-planes during Step 3.

Eq. (7) becomes Reynolds number independent. This independence ensures that the boundary conditions are well defined in the limit of very high Reynolds number considered in this work. Even though x may not be exactly on the wall itself, the shear stress τ_w is taken to represent the stress components $\tau'_{13}(x) = \tau'_{31}(x)$, and all other components of the stress tensor in the representation referred to the rotated coordinates (x'_1, x'_2, x'_3) are assumed to be zero. Specifically, $\tau'_{ij}(x) = \tau_w(\delta_{i1}\delta_{j3} + \delta_{i3}\delta_{j1})$. The use of a constant τ_w region of width δ (on the order of h) implies that this IB method "smears" the variables over lengths-scales comparable to the grid scale [48]. Finally, this stress tensor representation in the local coordinate system is transformed (rotated) to the global (x_1, x_2, x_3) (or (x, y, z)) coordinate system to set components of the stress tensor $\tau(x)$ used in the discretized LES equations. The tensor rotation is carried out using $\tau_{ij}(x) = a_{im}a_{jn}\tau'_{mn}(x)$, where the a_{ij} are the direction cosines between the global x_i-axis and the rotated x'_j-axis. The underlying assumption in choosing δ is that the velocity profile may be approximated by a log-law at distances less than δ from the wall. In cases where the logarithmic region of a boundary layer thickness is smaller than our choice $1.1h$, or where no logarithmic region exists, the approximation may introduce errors if skin-friction is important. However, as pointed out above, skin friction is not of primary importance for the geometries considered here (as shown in Appendix A), since pressure drag is the dominant source of drag on the square-cylinder branch elements.

Step 2. Extrapolate to obtain stresses at any point x in the band $-\delta \leqslant \varphi < 0$ just inside the IB by using the surrounding stresses outside the IB. Starting at x, a step of size ϵ is taken along the normal $\hat{n}$ to define a point in the fluid $\xi_1 = x + \epsilon\hat{n}$. Initially $\epsilon = h$ is used, but if this choice results in $\varphi(\xi_1) < 0$, then $\epsilon = 1.5h$ is used instead. A second fluid point along the normal is $\xi_2 = x + 2\epsilon\hat{n}$. The stress tensor is interpolated to get $\tau(\xi_1)$ and $\tau(\xi_2)$ using neighboring fluid points. The stress tensor $\tau(x)$ is extrapolated from these two fluid points into the solid region using $\tau(x) = 2\tau(\xi_1) - \tau(\xi_2)$.

Step 3. The last step is a stress-smoothing step that acts only on stresses further inside the IB, at any point x with $\varphi(x) < -\delta$. The stress field at these points is set by performing five successive overrelaxation (SOR) iterations [51] of the Laplace equation in τ on (x, y)-planes to obtain a smooth stress field that is amenable to the spectral differentiation used in these planes and blends smoothly with the stresses at points close the IB (from

the second step, above). Specifically, the stress at a point $x_{i,j,k}$ on the grid with $\varphi(x_{i,j,k}) < -\delta$ is calculated by performing the following update five times:

$$\tau_{i,j,k}^{m+1} = (1 - \omega_{\mathrm{SOR}})\tau_{i,j,k}^m + \omega_{\mathrm{SOR}}(\tau_{i-1,j,k}^m + \tau_{i+1,j,k}^m + \tau_{i,j-1,k}^m + \tau_{i,j+1,k}^m)/4, \tag{8}$$

where m keeps track of the iteration number (it is not related a physical time advancement), and the SOR parameter $\omega_{\mathrm{SOR}} = 1.5$ is used to accelerate the convergence. Tests show that the results are insensitive to increasing the number of SOR iterations performed.

Tests outlined in Appendix A show that the forces on an square cylinder with diameter to mesh ratio $d/h = 8$ can be accurately calculated using this approach. For the purpose of measuring forces, any branches with cross-section edge at least $8h$ in length are said to be resolved in this paper. The roughness length used in the log-law for the bottom wall and resolved tree surface is $z_0 = 10^{-4}L$. The effect of z_0 on the tree surfaces is small for the tree geometries used in the manuscript, since they are all constructed from square cylinders with sharp corners. With this geometry, pressure drag provides the overwhelming contribution to the total drag. This is verified in tests using flow over an isolated square cylinder, presented in Appendix A, which show that the effect of z_0 is small.

The simulation domain is a box defined by $\{(x, y, z): -1/2 \leqslant x/L \leqslant 3/2, -1/2 \leqslant y/L \leqslant 1/2, 0 \leqslant z/L \leqslant 1\}$, with a tree based at $(x, y, z) = (0, 0, 0)$ on the bottom wall. The top of the domain $(z = L)$ is treated as a stress-free, impenetrable boundary, and periodic boundary conditions are applied at the sides of the domain $(y = \pm L/2)$ to mimic the effect of an infinite row of trees. At the $x = -L/2$ plane, the time-varying velocity field from a precursor simulation of a fully-developed turbulent boundary layer is imposed. It uses the same equilibrium wall-layer boundary condition at the bottom surface and the stress-free boundary condition at the top. In the precursor simulations, no tree is present and flow is forced through the domain parallel to the x-axis by applying a constant mean pressure gradient forcing $f_\mathrm{p} = -\mathrm{d}\bar{p}/\mathrm{d}x > 0$ per unit volume, where $\bar{p}$ is the mean pressure. All flow variables are non-dimensionalized by ρ, L, and f_p, resulting in a time scale $\tau = (L\rho/f_p)^{1/2}$ and a velocity scale $u_* = (L f_p/\rho)^{1/2}$ (the mean friction velocity). These same scales are used to non-dimensionalize the flow variables in the simulations with the trees. In the main simulation domain, one quarter of the domain, $1 \leqslant x/L \leqslant 3/2$, is used as a fringe region [52,53] to force the velocity field back to its $x = -L/2$ value, which allows us to simulate non-periodic flow in the x-direction using pseudospectral numerics. Specifically, for $1 \leqslant x/L \leqslant 3/2$, the velocity is imposed according to $\tilde{u}(x, y, z) = \tilde{u}(L, y, z)(1 - w(x)) + \tilde{u}_\mathrm{p}(y, z)w(x)$, with $w(x) = \frac{1}{2}[1 - \cos(\pi(2x/L - 2))]$, where $\tilde{u}_\mathrm{p}(y, z)$ is the velocity from the precursor simulation at a fixed streamwise location. This way, we obtain the inlet velocity $\tilde{u}(-L/2, y, z) = \tilde{u}_\mathrm{p}(y, z)$ at every time step. Note that within the fringe region, this velocity has non-zero divergence. The use of a code with pseudospectral numerics in the horizontal directions along with our IB implementation and our fringe forcing technique results in a degradation of the spectral accuracy there. Due to the complex interplay of errors introduced by the two techniques, and since the code's accuracy in the bulk of the flow is not the primary objective of this paper, we refrain from attempting a rigorous analysis of the code's accuracy.

3.2. Results

A series of four BRS, using $g = 0, 1, 2, 3$ was performed, each using using $512 \times 256 \times 256$ computational grid points. Snapshots of the resulting velocity fields are shown in Fig. 3(a)–(d). The $g = 3$ simulation is under-resolved since the smallest branches are only $4h$ in diameter, so this simulation is included only to indicate a possible trend with the understanding that there could be some error in the results.

The resulting drag coefficients C_T from this series of BRS are summarized in Fig. 4. As expected, there is an increase in C_T associated with each generation of branches that is added. Plotted in log-scale, it appears that the sequence of C_T values would converge for g large enough, but this limiting value cannot be determined from these simulations without incurring uncontrolled extrapolation errors. The dependence of C_T on g deviates from that in Eq. (3) and indicates that higher values of g than those simulated here are required to demonstrate convergence of C_T. The relatively low values of C_T measured from these low-g tests compared to the simple estimate $C_T^{(\mathrm{ind})}(g \to \infty)$ above also indicate much larger values of g are necessary to achieve convergence to the asymptotic value.
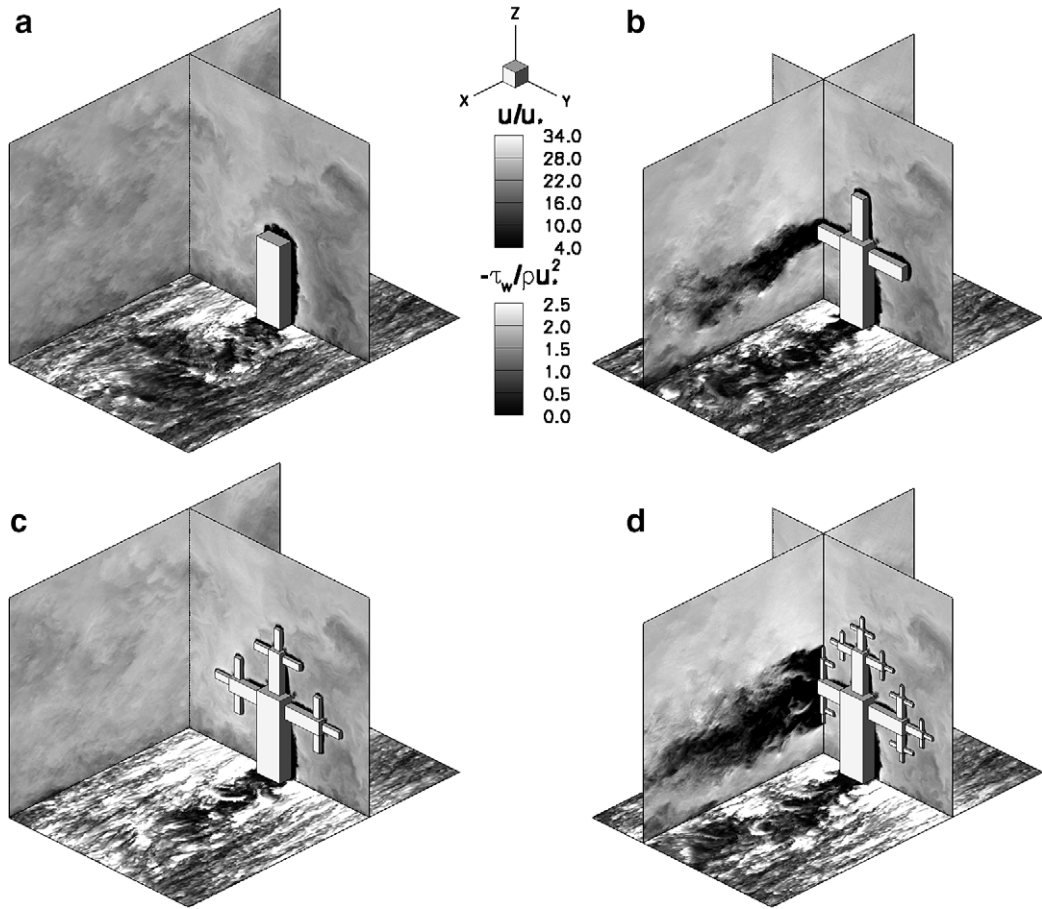
Fig. 3. Resolved tree geometry and instantaneous streamwise (x) velocity-field slices from $512 \times 256 \times 256$ BRS that resolve the trunk and g generations of sub-branches: (a) $g = 0$, (b) $g = 1$, (c) $g = 2$, and (d) $g = 3$. The bottom surface shows the instantaneous wall shear stress. To illustrate the wake structure, the (x,z)-planes in (b) and (d) are chosen so that they cut through the tree and the wake it produces.
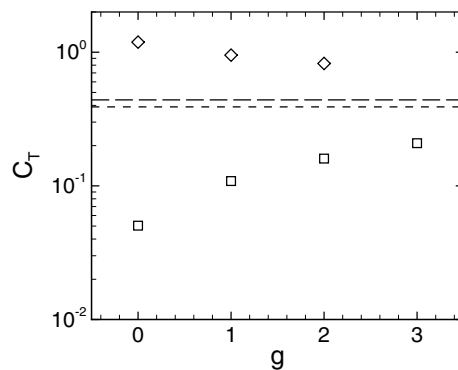

Fig. 4. Total drag coefficient C_T versus number of branch generations g from $512 \times 256 \times 256$ BRS (squares), BRS with the geometry augmented by plates (diamonds), and a priori estimate (dashed line). The long-dashed line shows the result to be described in Section 4.2, obtained via RNS.

To place an upper bound on the limiting value of $C_T(g \rightarrow \infty)$, a different set of simulations is performed. In these simulations, the asymptotic fractal tree is approached from a construction in which all branches smaller than generation g branches are represented by solid "plates", as shown in Fig. 5. Simulations were performed
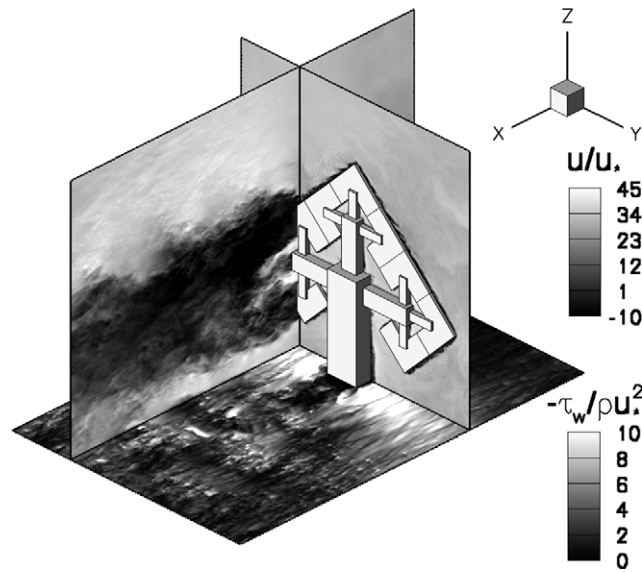
Fig. 5. Instantaneous velocity from simulation with $g = 2$, augmented by plates. The bottom surface shows the instantaneous wall shear stress.

for $g = 0$, 1 and 2 using this solid-plate representation of the sub-grid branches. The solid diamond-shape plates have thickness equal to the diameter of the next generation $(g + 1)$ branches. The same immersed boundary implementation described in Section 3.1 is used to impose equilibrium log-law and no penetration boundary conditions. In the limit $g \rightarrow \infty$, geometrically speaking, the tree tends to a situation very similar to that shown in Fig. 1, with some possible subtle differences at the very smallest scales where the solid-plate geometry will contain a connected border line, whereas the original construction consists of disjoint points that tend towards each other along this border line. As can be seen in Fig. 4, where diamonds show the resulting drag coefficients as a function of g, the resulting drag coefficient indeed falls above the ones from the earlier approach. The results are consistent with a tendency towards a common asymptotic value, but it is not possible to conclude this with any degree of certainty from these finite-size simulations.

These results highlight the need for a methodology to model the effects of unresolved branches and avoid resolving all the branches explicitly. Such a technique is proposed in the next section.

4. Renormalized numerical simulation (RNS)

RNS is an inverse-renormalization, down-scaling strategy. It solves the large-scale problem via numerical simulation while simultaneously learning from this simulation how to model the small-scale problem. This information about the small-scale problem is repeatedly fed back into the simulation of the large-scale problem, as it progresses in time, to achieve a simulation in which the effects of many small scales are well-represented even though they are not explicitly calculated. Dimensional analysis is applied to the resolved-scale features of a simulation, effectively down-scaling (shrinking) the quantities of physical interest so that they may be used to model the unresolved scales.

4.1. Formulation

Since the flow and tree interact through pressure and viscous forces that develop at the tree boundary, the effects of the tree on the flow may be viewed as the result of application of a force field $f(x)$ to the fluid, which is taken to be defined per unit volume. This concept of representing a solid body in a fluid only by the force it applies to the fluid is the basis of immersed boundary methods [46–49]. In the present study, RNS requires that the trees be partially resolved, i.e., some branches are large enough to be represented directly in a numerical

simulation, without a parameterization. Since in the present application the immersed boundary method is used to represent the resolved branches, the immersed boundary force field $r(x)$ (without the time index, for convenience) is known in the interior of the trunk and resolved branches of the simulation. From this point on, let the last such resolved branch generation be denoted by g. The total force field implied by the full tree object is written as

$$f(x) = r(x) + \sum_{\beta} f_{\beta}(x), \tag{9}$$

where the sum is over all unresolved branches β at generation $g + 1$, and $f_{\beta}(x)$ is a force field representing effects of these branches and all higher-generation descendants of the fractal attached to them. The sketch in Fig. 6 illustrates these variables. These force fields are defined such that they vanish outside the object they represent, e.g., $r(x) = 0$ for x falling outside the resolved trunk or branches, and $f_{\beta}(x) = 0$ for x falling outside the region covered by the branch β and its descendants (although on a finite computational mesh, this can only be approximately enforced, see Eq. (16)).

The methodology distinguishes between the distributed force fields applied by a given branch plus its descendants (e.g., $f_{\beta}(x)$) and the total force applied by this branch and its descendants,

$$F_{\beta} = \int f_{\beta}(x)\,\mathrm{d}^3x. \tag{10}$$

Equivalently, for generation g the total force due to a branch b and its descendants is given by

$$F_b = \int \left[r_b(x) + \sum_{\beta \in \mathrm{sub}(b)} f_{\beta}(x) \right] \mathrm{d}^3x, \tag{11}$$

where r_b is the immersed boundary force field due to b, and $\mathrm{sub}(b)$ is the set of generation-$(g + 1)$ sub-branches attached to b, i.e., $\{w_{b_g} \circ \cdots \circ w_{b_1} \circ w_i(T) : i = 0, \ldots, N_B - 1\}$.

The difficulty, and primary focus of RNS, is the determination of the force fields $f_{\beta}(x)$ at positions corresponding to the unresolved branches at generations $g + 1$ and above. The essential feature of RNS is to relate the total force F_{β} on the fluid due to branch β and its descendants to the total force due to the generation g branch $b = w_j^{-1}(\beta)$ and its descendants (and possibly other identical branches based at generation g).

At time step n, the total force F_{β}^n is modeled using a drag law of the form

$$F_{\beta}^n = -c_D^n(g + 1)\frac{\rho}{2}|V_{\beta}^n|V_{\beta}^n A_{\beta}, \tag{12}$$
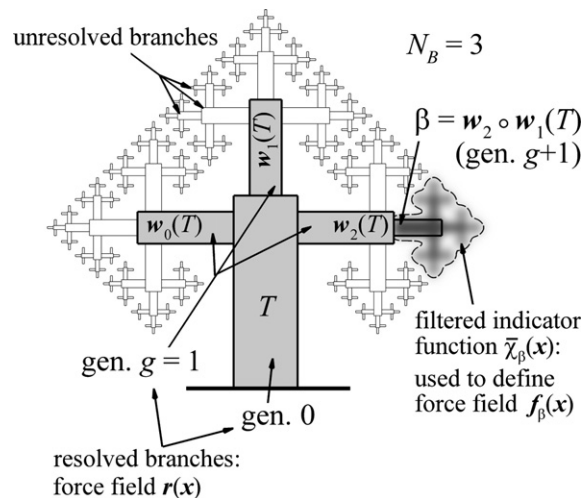

Fig. 6. Schematic diagram illustrating variables for RNS procedure for the case $N_B = 3$, $g = 1$. The smoothed indicator function $\bar{\chi}_{\beta}$ is shown for the branch $\beta = w_2 \circ w_1(T)$.

where $c_D^n(g+1)$ is the drag coefficient of β (and its descendants), V_β^n is the spatially averaged velocity vector in a region R_β in the general neighborhood of β and its descendants, and $A_\beta = ld/(1 - N_B r^2)$ is the projected area of branch β and its descendants as seen by the oncoming mean flow. We choose R_β to be the volume upstream of the branch, containing fluid that will hypothetically come into contact with the resolved and unresolved branch region. In this application, R_β is an upstream diamond shape as shown in Fig. 7. If written for the trunk generation, the upstream diamond shape is given by $R_T = \{(x, y, z): -W/4 \leqslant x \leqslant 0, |y| + |z - l_0| \leqslant W/2\}$, where $W = d_0 + 2rl_0/(1 - r)$. As documented in detail in Appendix B, other choices of R_β can influence the validity of key assumptions to be made below. The model in Eq. (12) assumes the direction of the force F_β^n to be opposite that of the velocity V_β^n, and neglects components of the force in other directions. However, the proposed methodology can be generalized to deal with other force components, e.g., lift and axial forces.

The drag coefficient $c_D^n(g+1)$ in Eq. (12) is unknown because of the complexity of the flow around the branches, but in incompressible flow, $c_D^n(g+1)$ can depend only on Re and non-dimensional parameters describing the flow geometry. Here, we assume Re is high enough that $c_D^n(g+1)$ is Re-independent. (Using square cross-section branches provides a firm basis for this assumption, since the drag coefficient of an isolated square cylinder in incompressible flow is essentially Re-independent for $Re > 2 \times 10^4$ [54].) This leaves only non-dimensional combinations of flow geometry parameters as possible variables on which the drag coefficient depends.

Since the tree geometry is self-similar, many of these non-dimensional geometric parameters will be the same (or nearly so) for different branch groups, even if the branch groups are based at different generations. In the present application, all branches at a given generation are identical and orthogonal to the incoming flow, so coefficients $c_D^n(g+1)$ corresponding to different branch groups based at generation $g+1$ are assumed to be equal, i.e., we neglect the dependence of $c_D^n(g+1)$ on the orientation of the branch group within the plane perpendicular to the incoming flow.

If there exists dependence of $c_D^n(g+1)$ on g (i.e., scale-dependence) it would arise through non-dimensional geometric parameters that are different for branches at different generations. An example is the possible dependence of c_D upon the detailed shape of the local velocity profile impinging on the branches (not just the mean velocity), which can depend on the position of the branches within the domain (and hence scale), e.g., through the height above the lower wall. As shown in Appendix B, the importance of the scale dependence on this local velocity can be made small by the choice of an appropriate local velocity averaging region R_β. As an approximation in this first application of RNS, we neglect these scale-dependent geometrical dependencies, and assume that $c_D^n(g+1)$ is independent of g, i.e., scale-independent.
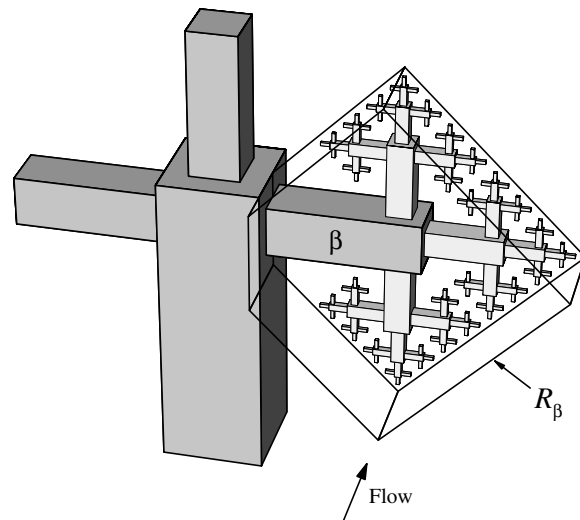

Fig. 7. Upstream diamond box used to specify the region of velocity averaging R_β in the definition of V_β.

Consequently, $c_D^n(g+1)$ can be measured from the branches at generation g. With the above assumptions, the problem is reduced to that of determining a single drag coefficient $c_D^n(g)$ that is to be measured using information from branches at generation g and is then to be applied to parameterize branches at generation $g+1$.

To measure $c_D^n(g)$, the model in Eq. (12) is applied to a generation-g branch b: $\boldsymbol{F}_b^n = -c_D^n(g)\rho|\boldsymbol{V}_b^n|\boldsymbol{V}_b^n A_b/2$. This acts as a constraint on $c_D^n(g)$, since another estimate for $\boldsymbol{F}_b^n$ is available from Eq. (11). From Eq. (11), we see that $\boldsymbol{F}_b^n$ includes both the resolved force due to branch b as well as the modeled forces from its unresolved descendants. The latter are calculated using the value of $c_D^{n-1}(g+1)$ that is known from the previous step of this iterative procedure. To make this dependence explicit, we write $\boldsymbol{F}_b^n = \boldsymbol{F}_b^n(c_D^{n-1})$. Specifically, the integrated form of Eq. (11) is used as

$$\boldsymbol{F}_b^n(c_D^{n-1}) = \boldsymbol{R}_b^n - \sum_{\beta \in \mathrm{sub}(b)} c_D^{n-1}(g+1)\frac{\rho}{2}|\boldsymbol{V}_\beta^n|\boldsymbol{V}_\beta^n A_\beta, \tag{13}$$

where $\boldsymbol{R}_b$ is the total resolved force due to b.

Since for $g \geqslant 1$ there will be more than one resolved branch b at generation g, and also since $\boldsymbol{F}_b^n$ and $\boldsymbol{V}_b^n$ need not be collinear, a least-squares approach is used to determine $c_D(g)$. This involves minimizing the total square error ε from all (resolved) branches at generation g and force vector components:

$$\varepsilon = \sum_{\mathrm{gen}(b)=g} \left[\boldsymbol{F}_b^n(c_D^{n-1}) + c_D^n(g)\frac{\rho}{2}|\boldsymbol{V}_b^n|\boldsymbol{V}_b^n A_b \right]^2. \tag{14}$$

The summation range $\mathrm{gen}(b) = g$ signifies that the sum includes every branch b at the last resolved generation g. Solving for the unknown average drag coefficient yields

$$c_D^n(g) = -\frac{2\sum_{\mathrm{gen}(b)=g}[\boldsymbol{F}_b^n(c_D^{n-1}) \cdot \boldsymbol{V}_b^n]|\boldsymbol{V}_b^n|A_b}{\sum_{\mathrm{gen}(b)=g}\rho|\boldsymbol{V}_b^n|^4 A_b^2}. \tag{15}$$

This value of $c_D^n(g)$ determined here at generation g is used in determining the magnitude of the forces $\boldsymbol{F}_\beta^n$ representing branches of generation $g+1$ and higher by assuming scale-invariance, i.e., $c_D^n(g+1) = c_D^n(g)$. By iteratively calculating $c_D^n(g)$ (in time), the force due to a large number of unresolved branch generations can be argued to have been included. The iterations in Eq. (15) at the start of a simulation are initialized using $c_D^0(g+1) = 0$. With the drag coefficient $c_D^n(g+1)$ we may determine the total force on the flow due to an unresolved branch β and its descendants.

We now turn to the issue of how to distribute this force on the computational mesh. Maintaining the inertial scaling of forces, we assume the force on the fluid at point $\boldsymbol{x}$ due to unresolved branches depends quadratically on the local velocity (and opposes it):

$$\boldsymbol{f}_\beta^n(\boldsymbol{x}) = -k_\beta^n|\tilde{\boldsymbol{u}}^n(\boldsymbol{x})|\tilde{\boldsymbol{u}}^n(\boldsymbol{x})\bar{\chi}_\beta(\boldsymbol{x}), \tag{16}$$

where k_β^n is a dimensional coefficient to be determined later, and where $\bar{\chi}_\beta$ is a smoothed version of the branch indicator function χ_β. In principle, this indicator function is 1 inside β and its descendants, and is 0 elsewhere. Since this cannot be represented on the finite grid, it is spatially filtered using a truncated Gaussian filter with filter radius $2h$ and truncation radius $4h$ (where h is the mesh spacing) to smooth it before sampling it on the mesh. Tests using different widths of this filter show the overall results to be insensitive to this choice of filter-scale. The filtered indicator function $\bar{\chi}_\beta$ is calculated by filtering an isolated copy of χ_β. This is done so that there is no cross-talk between branch β (and its descendants) and another nearby branch β' (and its descdents) during the filtering of χ_β which could lead to portions of β' showing up in $\bar{\chi}_\beta$. With this procedure forces from separate branches may be superimposed as in Eq. (9) without double-counting. What remains to be done is to relate the unknown (dimensional) coefficient k_β^n to the overall drag coefficient $c_D^n(g+1)$ such that the total given force from the distributed force of Eq. (16) equals the total force as implied by Eq. (12). Substitution yields the expression

$$k_\beta^n \int |\tilde{\boldsymbol{u}}^n(\boldsymbol{x})| \tilde{\boldsymbol{u}}^n(\boldsymbol{x}) \bar{\chi}_\beta(\boldsymbol{x}) \mathrm{d}^3\boldsymbol{x} - c_{\mathrm{D}}^n(g+1) \frac{\rho}{2} |\boldsymbol{V}_\beta^n| \boldsymbol{V}_\beta^n A_\beta = 0, \tag{17}$$

where $c_{\mathrm{D}}^n(g+1) = c_{\mathrm{D}}^n(g)$ is assumed to be known from Eq. (15). Since the above vector equation for scalar k_β^n is over-determined, again a least-squares error approach is used, leading to

$$k_\beta^n = c_{\mathrm{D}}^n(g+1) \frac{(\rho|\boldsymbol{V}_\beta^n|\boldsymbol{V}_\beta^n A_\beta/2) \cdot \boldsymbol{I}_\beta^n}{|\boldsymbol{I}_\beta^n|^2}, \tag{18}$$

where

$$\boldsymbol{I}_\beta^n = \int |\tilde{\boldsymbol{u}}^n(\boldsymbol{x})| \tilde{\boldsymbol{u}}^n(\boldsymbol{x}) \bar{\chi}_\beta(\boldsymbol{x}) \mathrm{d}^3\boldsymbol{x}. \tag{19}$$

The force $\boldsymbol{f}_\beta^n(\boldsymbol{x})$ thus represents structural information about the unresolved forces at scales between the size of R_β, all the way down to the grid scale h. Spatial information about scales finer than the grid resolution is not directly represented using the coarse grid. Instead, forces due to these scales are taken into account only through $c_{\mathrm{D}}^n(g+1)$, which spreads their influence out over the (discrete) support of $\bar{\chi}_\beta$ for branches β at generation $g+1$.

To apply Eq. (16), each branch β at generation $g+1$ is considered one at a time. All points $\boldsymbol{x}$ near branch β and its descendants are considered and the indicator function $\bar{\chi}_\beta(\boldsymbol{x})$ is calculated using a branch-local coordinate system. If $\beta = w_{\beta_{g+1}} \circ \cdots \circ w_{\beta_1}(T)$, the IFS is used to define the origin of the branch-local coordinate system as $\boldsymbol{O}_\beta = w_{\beta_{g+1}} \circ \cdots \circ w_{\beta_1}(\boldsymbol{O})$, with $\boldsymbol{O} = (0,0,0)$ denoting the origin of the global coordinates. The local coordinate unit vectors $\boldsymbol{e}_{j,\beta}$ for $j = 1, 2, 3$ are defined in the directions of $w_{\beta_{g+1}} \circ \cdots \circ w_{\beta_1}(\boldsymbol{e}_j) - \boldsymbol{O}_\beta$, where $\boldsymbol{e}_j$ is the unit vector in the x_j direction in the global coordinates (and $x_1 = x$, $x_2 = y$, $x_3 = z$). Since all the branches are identical, the indicator function $\bar{\chi}_\beta$ is the same for each branch when referred to the local coordinates, so only one copy of $\bar{\chi}_\beta$ relative to these coordinates is stored. Call this copy $\bar{\chi}$. To apply the force in Eq. (16), $\bar{\chi}_\beta$ is calculated by first transforming $\boldsymbol{x}$ into the branch-local coordinates $\boldsymbol{\xi}_\beta$ of branch β using $\boldsymbol{\xi}_\beta = [\boldsymbol{e}_{1,\beta} \boldsymbol{e}_{2,\beta} \boldsymbol{e}_{3,\beta}]^T (\boldsymbol{x} - \boldsymbol{O}_\beta)$, and then using these coordinates to calculate $\bar{\chi}_\beta(\boldsymbol{x}) = \bar{\chi}(\boldsymbol{\xi}_\beta)$ using the stored copy. When $\boldsymbol{\xi}_\beta$ does not fall on a point of the grid used to store $\bar{\chi}$, trilinear interpolation is used to calculate $\bar{\chi}(\boldsymbol{\xi}_\beta)$. However, if $\boldsymbol{x}$ lies within a resolved branch, then $\bar{\chi}_\beta(\boldsymbol{x})$ is set to zero, so the support of $\boldsymbol{f}_\beta(\boldsymbol{x})$ is disjoint from that of $\boldsymbol{r}(\boldsymbol{x})$.

The momentum sink provided by the RNS force also affects the turbulent kinetic energy budget. The body force (per unit volume) applied by RNS to the flow is given by Eq. (16), where we expect $k_\beta > 0$. This means that the RNS force dissipates resolved kinetic energy $\frac{1}{2}\tilde{\boldsymbol{u}} \cdot \tilde{\boldsymbol{u}}$ at the spatially-dependent rate $\varepsilon_{\mathrm{RNS}}(\boldsymbol{x}) = -\boldsymbol{f}_\beta \cdot \tilde{\boldsymbol{u}} = k_\beta \bar{\chi}_\beta(\boldsymbol{x}) |\tilde{\boldsymbol{u}}(\boldsymbol{x})|^3 \geqslant 0$.

In summary, the RNS procedure consists of writing the distributed force due to the fractal tree according to Eq. (9), where $\boldsymbol{r}(\boldsymbol{x})$ is given by the usual immersed boundary method to freeze the velocity at the points inside resolved trunks and branches. The distributed force fields $\boldsymbol{f}_\beta(\boldsymbol{x})$ inside unresolved branches are determined using Eq. (16) with the coefficient k_β^n as given by Eq. (18), and with $c_{\mathrm{D}}^n(g+1)$ $(= c_{\mathrm{D}}^n(g))$ determined from the basic RNS procedure according to Eq. (15). In Eq. (15), $\boldsymbol{F}_b^n$ is determined from Eq. (13), which in turn uses the values of c_{D}^{n-1} obtained from the previous time iteration.

The recursive relation Eq. (15) for $c_{\mathrm{D}}^n(g)$ is not unique. In the present (explicit) implementation we chose to iterate in time, so $c_{\mathrm{D}}^{n-1}(g)$ from the previous time-update is used to compute the sub-branch contribution to $\boldsymbol{F}_b^n$ on the right-hand side of Eq. (15). An alternative (implicit) approach would be to make the dependence of $\boldsymbol{F}_b^n$ on $c_{\mathrm{D}}^n(g)$ explicit during the error minimization, which is analogous to the dynamic approach [33,34] often used in turbulence modeling. Tests have shown that there is no significant difference between the two approaches in the current application although the proposed iterative method appears to be more stable in some other applications we have attempted, and so it is maintained here as well.

Generally, a drag coefficient by itself is not useful unless the area and velocity scale upon which it is based are also specified. Since our c_{D} is defined in terms of the velocity scale based on R_β, it is not directly comparable to drag coefficients based on other velocity scales, e.g., the free stream velocity. In fact, since R_β depends on r, c_{D} for trees with different r (but the otherwise similar geometry) are not directly comparable. With a velocity scale based on R_β, c_{D} can be viewed as an internal model parameter that provides a certain amount of drag to the flow corresponding to the tree. A measure of this drag is C_T, which is based on the average

velocity in the domain and the frontal area of the domain, so that it can be compared directly among different tree geometries and (with appropriate area conversions) drag coefficients from other studies. Therefore, we report most results in terms of C_T, and place less emphasis on specific values of c_D.

4.2. Application

As a basic test of the RNS methodology, it is used to simulate the flow from Section 3. The RNS is performed using a resolution of only $128 \times 64 \times 64$ grid points, resulting in a trunk diameter of $8h$, which is just sufficient to resolve the tree trunk. All other branches are unresolved and modeled using the RNS technique, i.e., $g = 0$. To further reduce CPU time, the RNS-determined $c_D(g = 0)$ is only updated once every five time steps. A snapshot of the RNS velocity field, with the RNS-determined force field for the unresolved branches superimposed, is shown in Fig. 8(a). Note that the force is active only in the regions where branches are located, and within these regions the force is non-uniformly distributed and smoothed. Mean velocity contours from the RNS are shown in Fig. 8(b). The resistive effect of the RNS-determined force, causing a lower velocity at position corresponding to the unresolved branches, is apparent.

The IB treatment does capture vortex shedding effects in simulations of flow over an isolated square cylinder (see Appendix A), e.g., the Strouhal number estimated from the peak in the lift spectrum gives a reasonably accurate answer. In the flow over the trees, some vortex shedding occurs near the lower part of the trunk. However, tree-scale vortex shedding is not observed due to the presence of the branches on the upper part of the tree, which tend to break-up large-scale organization of the flow.

The time-history of the RNS-calculated model coefficient $c_D(0)$ in Fig. 9 demonstrates that the approach rapidly converges (in time) to a statistically stable mean value of $\bar{c}_D(0) \approx 3.9$. The fluctuations in $c_D(0)$ are a result of the turbulent fluctuations of quantities used in the model, i.e., the resolved force on the trunk and the velocity scale. Averaging the total force obtained in this simulation yields a global drag coefficient (which includes the resolved trunk) of $C_T \approx 0.44$. This is comparable to, but 11% higher than, the infinite generation estimate $C_T^{(ind)} \approx 0.39$ from Section 2.2. On average, the unresolved force in this simulation makes up 79% of the total force on the tree, which highlights the importance of accurately modeling the unresolved branches when predicting the total tree drag.

A slight difference with the BRS is that RNS effectively downscales the z_0 used on the resolved branches in parameterization of the unresolved branches, i.e., the ratio z_0/l for the last resolved generation branches is imposed on the unresolved branches. To measure the sensitivity of the RNS approach to the parameter z_0 on the trunk surface (originally taken as $z_0 = 10^{-4}L$ in the base case), tests with two other values of z_0 were also performed: $z_0 = 5 \times 10^{-5}L$ and $z_0 = 2 \times 10^{-4}L$. These tests showed less than a 1% difference from the base case in the value of C_T.
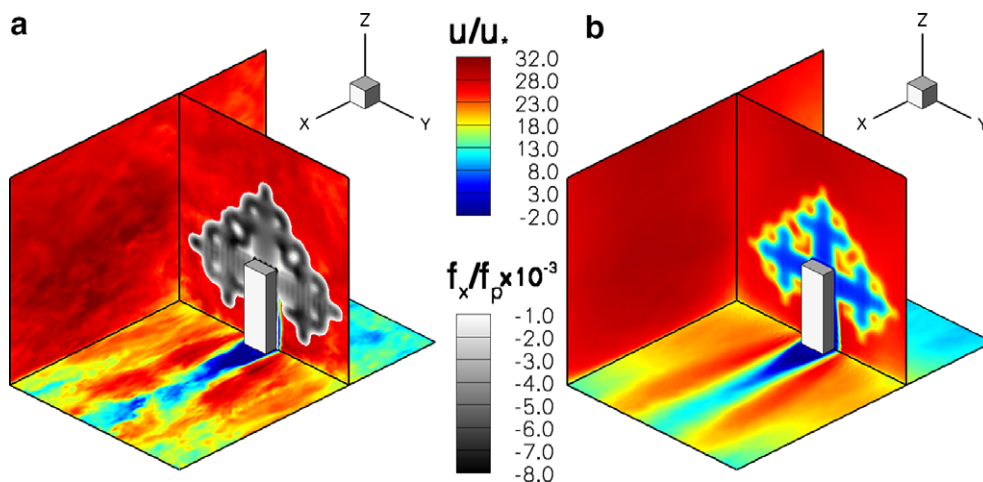

Fig. 8. Resolved tree geometry, and streamwise velocity-field slices from $128 \times 64 \times 64$ RNS with $g = 0$: (a) instantaneous streamwise velocity field (color), with streamwise force field superimposed (grey) and (b) mean streamwise velocity.
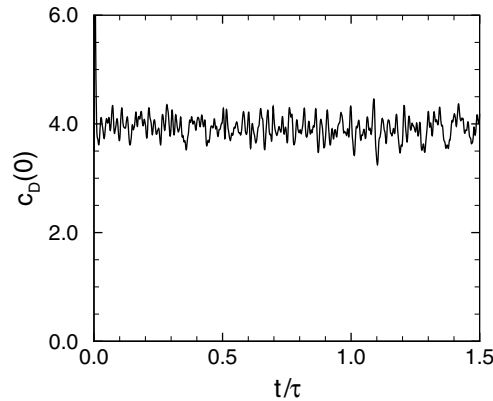
Fig. 9. Time history of the branch drag coefficient $c_D(g)$ from the RNS test with $g = 0$, showing that the procedure produces a statistically stable value.

Since the tree height $3L/4$ is a large fraction of the domain height L, domain-size effects do influence the drag on the tree. To test the sensitivity to domain-size, additional RNS simulations using domain heights $3L/2$ and $2L$ were performed. For the $3L/2$ and $2L$ simulations, the values of C_T were nearly equal, and about 5% lower than the value obtained using domain height L. As expected, C_T decreases when the domain height is increased to $3L/2$ due to the reduced confinement of the flow over the top part of the tree. The negligible difference observed in changing the domain height from $3L/2$ to $2L$ indicates that further increases in domain height would have only minor effects. Since the domain height effect is small, the simulations reported below continue to use domain height L.

We have also performed tests with different streamwise domain and fringe lengths. Simulations using $L_x/L = 2,3,4$ with fringe lengths of 12.5% and 25% of L_x show that C_T increases slightly with L_x, indicating that finite domain size does constrain the tree wake to some degree. Effects of the fringe size were found to be negligible.

4.3. Grid sensitivity tests

To quantify the sensitivity of C_T to the numerical resolution, a series of grid-resolution tests are presented. In the first test, flow over the $g = 2$ geometry (i.e., a tree trunk with two generations of branches) is simulated using mesh sizes $h_{64} = L/2^6$, $h_{128} = L/2^7$, and $h_{256} = L/2^8$. In this test, all branches are treated as resolved branches and no model is used for the unresolved branches, i.e., no RNS is used. In the Smagorinsky model $\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = -2c_S^2\Delta^2|\widetilde{S}|\widetilde{S}_{ij}$, the parameter Δ representing the implicit grid filter width is held fixed at $\Delta = h_{64}$ as the mesh spacing h is reduced. (The filtered variables $\tilde{u}_i$, $\widetilde{S}_{ij}$, etc., are still calculated assuming an implicit grid filter, so no explicit filtering operation is introduced even though $\Delta \neq h$.) A uniform inflow condition, with velocity $\boldsymbol{u} = (U_\infty, 0, 0)$ is used in this test. These grid sensitivity simulations using a laminar inflow can be thought of as having the trees standing almost entirely in the freestream (the flow develops a small boundary layer between the time in enters the domain and when it hits the tree). Note that the use of the Smagorinsky model with laminar inflow conditions may lead to inaccurate predictions of the transitional flow around and behind the tree, due to the excessive dissipation it provides. In this particular test, however, we are concerned only with the dependence of the numerical solution on mesh spacing (i.e., verification), and not whether the equations being solved numerically are a realistic physical description of the system (i.e., validation). The effective drag coefficients C_T for these three simulations are shown in Fig. 10 as a function of mesh spacing (represented by the dash-dot line with square symbols). It is evident that C_T is relatively insensitive to the mesh resolution. This shows that the resolved geometry in the other simulations presented here is sufficiently resolved to yield results insensitive to the resolution. The decrease in C_T when going from $h = h_{64}$ to $h = h_{128}$ is a consequence of the $g = 2$ geometry being significantly under-resolved at $h = h_{64}$, having only $2h$ resolution for the smallest branches.
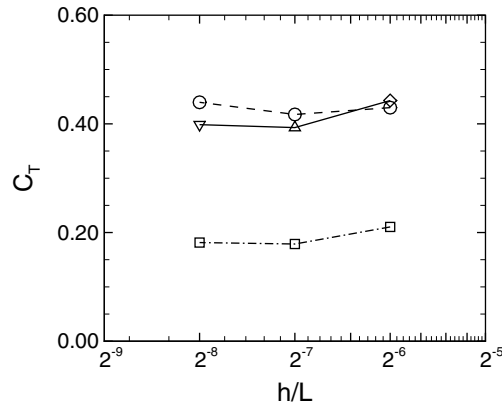
Fig. 10. Convergence tests showing time-averaged effective total drag coefficient C_T versus computational mesh spacing. Squares/dash-dot line: no RNS, trunk plus two branch generations, uniform inflow, $\Delta = L/2^6$. Circles/dashed line: RNS with $g = 0$, uniform inflow, $\Delta = L/2^6$. Solid line: RNS with $g = 0$ (diamond), $g = 1$ (up-pointing triangle), $g = 2$ (down-pointing triangle), turbulent inflow, $\Delta = h$.

The second grid-resolution test is similar to the first test, except that RNS is used, and only the trunk is treated as resolved, i.e., $g = 0$. The results of the test are also shown in Fig. 10 (as the dashed-line with circle symbols), and again C_T does not show large sensitivity to grid-resolution. This shows a coarse mesh may be used with RNS, and only minor changes are obtained by performing more finely resolved (and much more expensive) simulations.

In the final grid resolution test, the tree geometry is refined along with the mesh so the $8h$ resolution rule is always satisfied. This test thus also serves as test of the robustness of RNS to different levels of branch resolution. The simulations use $g = 0$, $g = 1$, and $g = 2$ in simulations with h_{64}, h_{128}, and h_{256} mesh sizes, respectively. The grid filter size Δ is also refined in this series and it is kept equal to the mesh spacing h. For this series, turbulent inflow conditions are specified from precursor simulations. The results from this series are shown as the solid line in Fig. 10, with different symbols to indicate the different resolved geometry in each case. The results show that the RNS with the resolved geometry refinement are also relatively insensitive to the mesh resolution used. Therefore, predictions made using the RNS approach are self-consistent, since the prediction of the quantity of physical interest C_T does not depend strongly on the level of resolution of the resolved geometry.

The two h_{64} RNS simulations with $g = 0$, one using laminar inflow, one using turbulent inflow, result in similar values of C_T. This is in contrast to the drop in drag coefficient observed in isolated square cylinder flow upon introduction of freestream turbulence. The reason for this is the effect of the turbulence on the vortex shedding [55]. Since our tree geometry does not exhibit tree-scale vortex shedding (and only shedding around the base of the trunk), it is plausible the turbulence level does not have a large effect on C_T, as shown in our results.

4.4. Predictions

In this section, predictions from several RNS runs performed using a range of tree fractal dimensions, obtained by varying the scale ratio r and number of sub-branches N_B, are presented. Two flow configurations are considered: developing flow over a single row of trees (as in previous sections, denoted here by "D") and flow over many rows of trees (i.e., a square lattice with fully developed flow, denoted by "FD"). The purpose is to show the versatility of the RNS approach, in that it can adjust to a variety of flows and tree geometries. All the RNS in this section use $g = 0$, so the resolved geometry is identical in all cases.

In the first series of simulations (D-plus), flow over a row of trees with the $N_B = 3$ geometry considered in the preceding sections is performed, but the scale ratio is varied over the range $r = 0.20, 0.25, \ldots, 0.50$. Turbulent inflow from a precursor simulation is prescribed at the plane $x = -L/2$. The resulting C_T from this series are represented in Fig. 11(a) by square symbols, plotted as a function of the tree fractal dimension D. The increase of C_T with D is significant. The second simulation uses (FD-plus) the same tree geometry, but uses periodic streamwise boundary conditions with no fringe region. The simulation domain is changed to a cube with edge length L, which corresponds to a square lattice arrangement of trees on the plane $z = 0$ with
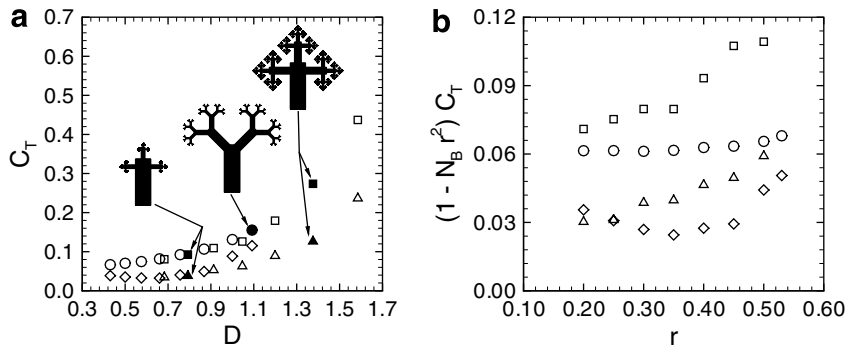
Fig. 11. Total drag coefficient C_T for two types of tree ("plus" and "Y") in two arrangements. The first is a single row of trees (squares for the D-plus and circles for the D–Y case). The second is a square-lattice canopy leading to a fully developed flow (triangles for the FD-plus case and diamonds for the FD-Y case): (a) shows C_T as a function of fractal dimension D and includes schematics of the corresponding frontal tree geometry. The solid symbols indicate the data corresponding to these schematics and (b) shows C_T premultiplied by a scaling-factor, as a function of scale ratio r.

a spacing L between the trees in both the x and y directions. To force the mean flow in the streamwise x direction, a constant mean pressure gradient force f_p is imposed, as described in Section 3.1. The results from this fully developed series with the $N_B = 3$ trees, are represented by the triangle symbols in Fig. 11(a). The C_T values from FD-plus fall below those from D-plus because the wake of the upstream trees slows the fluid impacting the trunk and low generation branches significantly, resulting is less drag force. A lower drag coefficient is the expected behavior for fully developed flow.

The third series of simulations (D–Y) consists of developing flow over a row of trees with $N_B = 2$ and the branches inclined to form a "Y" shape. For this geometry, the IFS is given by Eq. (2) with $R_0 = R_x(\pi/4)$, $R_1 = R_x(-\pi/4)$, $s_0 = (0, -d_0/(2\sqrt{2}), l_0 - d_0(2 - \sqrt{2})/4)$, and $s_1 = (0, d_0/(2\sqrt{2}), l_0 - d_0(2 - \sqrt{2})/4)$. The region R_β used for averaging the velocity, is defined for the trunk as $R_T = \{(x, y, z): -W/4 \leqslant x \leqslant 0,$ $|y| \leqslant y_{max}, z_{min} \leqslant z \leqslant z_{max}\}$, with $y_{max} = (rl' + d_0/2)(r + 1/\sqrt{2})/(1 - r^2)$, $z_{max} = l' + y_{max}$, $z_{min} = l' + (rl' + d_0/2)(1 - r^2)/\sqrt{2} - r^3 d_0/2 - r^4 z_{max}$ and $l' = l_0 - d_0/2$. In the Y-series, the additional value $r = 0.53$ is added to the range of r considered before for the plus series. This series is otherwise identical to D-plus. The results of this test are represented by the circles in Fig. 11(a). The results from tests D–Y and D-plus show good collapse over the range of D tested.
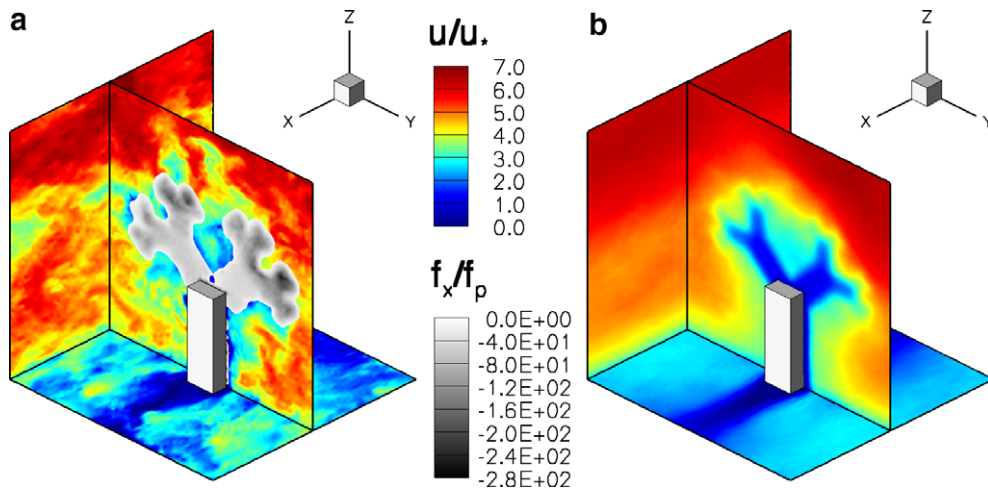

Fig. 12. (a) Instantaneous streamwise velocity field, with streamwise component of force field superimposed (grey scale), from coarse (64^3) RNS using the Y geometry, periodic boundary conditions on the lateral boundaries, and resolving only the tree trunk. (b) Mean streamwise velocity.

The last series is with the Y trees and periodic, fully-developed conditions (FD-Y). Fig. 12(a) shows slices through an instantaneous streamwise velocity field and RNS force field, and Fig. 12(b) shows slices through the time-mean streamwise velocity field. Although the velocities should not be compared directly, the relative intensity of the turbulence in Fig. 12(a), from the FD-series, is larger than that in Fig. 8(a), from the D-series, because the incoming wake turbulence from upstream trees in the FD-series is more intense than the turbulence developed in the precursor simulation that provides the inflow for the simulations in the D-series. Results are represented by the diamond symbols in Fig. 11(a), which shows that this series yields lower drag than D–Y (again, due to the wake of upstream trees), but higher drag than FD-plus. The results from FD-Y and FD-plus do not exhibit the same collapse that was observed in the D–Y and D-plus cases, and instead show different dependence on D, with C_T values from FD-Y growing faster with D than those from FD-plus.

Finally, the results from this section are re-plotted as a function of scale ratio r in Fig. 11(b). In this plot, C_T has been premultiplied by the asymptotic area ratio $1 - N_B r^2$ to facilitate comparison with the prediction of Eq. (3). In the $g \rightarrow \infty$ limit, Eq. (3) predicts $(1 - N_B r^2)C_T = C_{sq}l_0 d_0/L^2 \approx 0.097$ for each tree, which means all of the simulations should fall on the same horizontal line. The results for each case do not fall on horizontal lines (except perhaps the D-Y case, which however does not fall on the predicted horizontal line), suggesting that RNS is sensitive to differing strengths of interactions that occur among branches when r and N_B are varied. Plotted this way, the C_T from the two Y-series are always lower than the corresponding plus-series values for a given r because there are fewer branches to resist the fluid motion.

5. Conclusions

Turbulent flow over arrangements of idealized fractal trees has been studied via a series of numerical simulations. High-resolution, branch-resolving LES applied at varying branch numbers show that the dependence of the drag coefficient on the inner cutoff scale of the fractal is quite strong, at least at low and moderate numbers of branch generations. It has also been shown that the speed of convergence to an asymptotic, generation-independent, drag coefficient is very slow.

To avoid great computational expense associated with directly simulating such multi-scale systems, a new technique, RNS, has been proposed. By exploiting geometric scale-invariance, it allows for the efficient computation of fluid interactions with fractal boundaries. The RNS technique makes predictions about the physical effects of the full range of scales in a system by using information from a limited, but affordable, range of large scales. The technique has been used to determine the effective drag coefficient over idealized fractal tree geometries, and the resulting predictions have been shown to be reasonable in three ways: they are within bounds set by expensive branch-resolving LES (i.e., simulations not using the RNS technique), they are self-consistent (insensitive to mesh and tree geometry resolutions), and they are within 11% of an a priori analytical estimate. Finally, the technique has been applied to study how the drag on fractal trees depends on the scale ratio r between branch generations. The drag is found to scale roughly with the frontal area of the fractal, but also depends in non-trivial ways on the detailed tree geometry and flow configuration.

The present results have been obtained by limiting the application to high Re flows, and under the assumption that $c_D(g)$ tends to a finite, non-zero limit as $Re \rightarrow \infty$ (i.e., complete similarity). In applications involving finite Re, this assumption could be weakened to allow for further determination of parameters associated with possible Re-dependencies. Perhaps considerations of incomplete similarity [56] in $c_D(Re)$ as a function of Re may be useful. Present results lead us to conjecture that many other physical systems displaying fractal geometric character, such as flow in arterial and pulmonary networks, flow through fractal porous media (such as soils) and turbulent atmospheric flow over fractal mountain ranges, could be modeled accurately using RNS at affordable computational cost.

Acknowledgements

Appendix A. Resolution requirements

Here, the minimum resolution requirements of the immersed-boundary branch treatment described in Section 3.1 are summarized. Simulations of an isolated square cylinder with edge length d in a uniform laminar inflow at $Re = 2.2 \times 10^4$ were performed. The reason for using a finite Re as opposed to considering $Re \to \infty$, as in simulations in the main body of this work, is that this allows direct comparison with experiments. It is important to note, however, that for bluff body shapes with sharp corners that fix the separation points (including the square cylinders throughout this study), the Re effects are small. This was shown experimentally for square cylinders in [54] (and this data have been quoted in more recent texts, such as [57]), which showed that the drag coefficient is Re-independent to a good approximation when $Re > 2 \times 10^4$, as long as the flow may still be considered incompressible. From this, we conclude that the gross flow features must be approximately independent of Re in this Re range. Therefore, the resolution requirements for $Re > 2 \times 10^4$ (i.e., for the simulations in the main body of this work) are roughly the same as those for $Re \sim 2 \times 10^4$, at least for the purpose of measuring the drag.

The dimensions of the flow domain were $L_x = 32d$, $L_y = 16d$, $L_z = 4d$, and a fringe region of length $4d$ was used. The roughness length of the cylinder was prescribed as $z_0 = 10^{-4} L_z = 4 \times 10^{-4} d$. Three regular Cartesian grids with equal grid spacing in each direction were used. The coarsest mesh consisted of $N_x \times N_y \times N_z = 128 \times 64 \times 16$ grid points, resulting in $d/h = 4$, where h is the grid spacing. The mid-resolution mesh had $N_x \times N_y \times N_z = 192 \times 96 \times 24$ grid points, and $d/h = 6$. The finest mesh presented here had $N_x \times N_y \times N_z = 256 \times 128 \times 32$ grid points, and $d/h = 8$. The uniform inflow U_∞ was parallel to the x-axis, and the cylinder axis was parallel to the z-axis. Impenetrable, stress-free boundaries were imposed at z-boundaries and periodic boundary conditions were imposed at y-boundaries. The cylinder was centered $5d$ downstream of the inflow plane. The mean drag coefficient $\overline{C}_D$ for the $d/h = 4$ grid was $\overline{C}_D = 2.80$, while it was $\overline{C}_D = 2.58$ for the $d/h = 6$ grid. These values are significantly larger than the experimental values (see Table 1). Results obtained using $d/h = 8$ are closer to experimental values and are summarized in Table 1 in terms of the mean drag coefficient $\overline{C}_D$, the r.m.s. drag coefficient fluctuations C'_D, the r.m.s. lift coefficient fluctuations C'_L, the Strouhal number St, and the normalized distance from cylinder center to the end of the recirculation region behind the cylinder x_r/d. Results from experimental investigations with nearly laminar inflow are also shown in Table 1 for comparison. In [39], $Re = 2.1 \times 10^4$ is used, while in [55] $Re = 1.76 \times 10^5$ is used. However, the quantities reported are known [58,59] to be insensitive to Re for $Re > 2 \times 10^4$. The overall agreement between the present LES and experiments is good, but the largest discrepancy is in C'_L, which is known to be sensitive to the turbulence level of the inflow. Indeed, the simulation value of $C'_L = 0.89$ does fall within the range 0.59–1.23 reported in [55] corresponding to a range of inflow turbulence intensities. See [60,61,49] for other LES studies of flow over a square cylinder. Note that the corners of the square cross-section essentially fix the flow separation points, allowing the use of a low number of grid points in the boundary layer around the cylinder. For smooth cross-section shapes, e.g., circles, higher resolution is required to accurately predict the separation points and hence the drag. In light of the results in Table 1, we conclude that forces on the cylinder are predicted with reasonable accuracy when $d/h = 8$.

A difference between the experiments and the simulations presented here is that the experimental cylinders have smooth surfaces, while the roughness length in the simulations is $z_0 = 4 \times 10^{-4} d$. To verify that the results are insensitive to z_0 (since form drag and not skin friction dominates the drag), simulations using two other values of z_0 were performed, and the drag coefficient statistics were compared. The resolution was $d/h = 8$ for these tests. Starting with an initial condition from a simulation with $z_0 = 4 \times 10^{-4}\ d$, simulations with the two additional values $z_0 = 2 \times 10^{-4} d$ and $z_0 = 8 \times 10^{-4} d$ were performed during a non-dimensional time

Table 1
Comparison of simulation and experiments for isolated square-cylinder flow

Source	$\overline{C}_D$	C'_D	C'_L	St	x_r/d
Present LES	2.23	0.23	0.89	0.140	1.33
Ref. [39]	2.1	–	–	0.132	1.4
Reference [55]	2.05	0.23	1.23	0.122	–

$tU_\infty/d = 80$. Averaging only over the last half of this time interval, the mean drag coefficient using $z_0 = 2 \times 10^{-4}$ differed from the base value by 1.3% (the base value being that obtained using $z_0 = 4 \times 10^{-4}$), while the drag coefficient obtained using $z_0 = 8 \times 10^{-4}$ differed from the base value by 1.7%. Thus, as expected, the mean drag shows very weak dependence on z_0.

Appendix B. Model velocity scale

In this appendix, four definitions of the region R_β used to obtain the velocity scale V_β corresponding to branch β and its descendants are compared. The consistency of the scale-invariance assumption of the model parameter $c_D(g)$ is shown to depend significantly on the choice of R_β.

The first three choices of R_β for any generation-$g + 1$ branch $\beta = w_{\beta_{g+1}} \circ \cdots \circ w_{\beta_1}(T)$ can be obtained from the region R_T corresponding to the trunk by $R_\beta = w_{\beta_{g+1}} \circ \cdots \circ w_{\beta_1}(R_T)$. These choices are illustrated in Fig. 13. The first choice of R_β is a large bounding box chosen around β that is guaranteed to contain β and its descendants, independent of how the branches are oriented. Such a box corresponding to the trunk is $R_T = \{(x, y, z): |x| \leqslant W/2, |y| \leqslant W/2, 0 \leqslant z \leqslant H\}$, with $W = d_0 + 2rl_0/(1 - r)$, $H = l_0/(1 - r)$. The second choice of R_β is the upstream part of the bounding box in the first choice. In terms of R_T, this is $R_T = \{(x, y, z): -W/2 \leqslant x \leqslant 0, |y| \leqslant W/2, 0 \leqslant z \leqslant H\}$. The third choice is the upstream diamond shape already described in Section 4.1. The final choice is the same for all the branches: it is simply the entire simulation domain.

These four velocity scales are tested at three tree geometry resolutions $g = 0, 1, 2$, with mesh spacings $h = L/2^6, L/2^7, L/2^8$, respectively. The SGS model filter width $\Delta = h$ is used in each test. The inflow conditions are a uniform inflow with velocity $\boldsymbol{u} = (U_\infty, 0, 0)$. The ratio $\bar{c}_D(g)/\bar{c}_D(0)$ (with the overbar denoting a time average) is shown in Fig. 14, to emphasize the scale-dependence of results obtained in each case. The bounding box velocity scales (the first two choices) show significant scale dependence, which indicates RNS with these velocity scales is not self-consistent. In contrast, the diamond velocity scale shows little scale dependence, indicating self-consistency when used in RNS with the present geometry. Finally, the domain-mean velocity scale, although yielding less scale-dependence than the bounding box choices, shows more scale dependence than the diamond scale. Since it exhibits the least scale dependence, as well as the desirable property of locality (which may be essential in more complex flow configurations), the diamond velocity scale is chosen for the present RNS application.

From these tests, we draw the following heuristic rules to help to define R_β. First, R_β should be chosen upstream of β (relative to the mean flow), in an attempt to separate cause and effect. Second, R_β should not contain large volumes of fluid that will not come in contact with β and its descendants. A guideline that helps enforce this second rule is to choose the boundary of R_β to pass through the branch tips of the generation-∞ branches. Of course, an additional constraint is that R_β be a relatively simple shape, for implementation purposes.
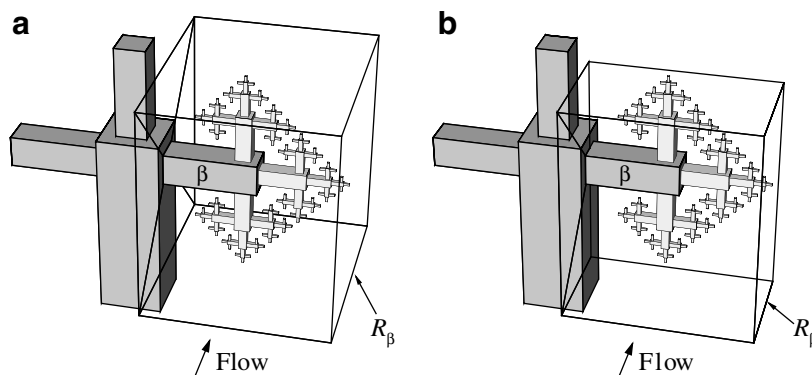

Fig. 13. Boxes used to define velocity scale V_β around branch β and its descendants. (a) bounding box, (b) upstream half bounding box. The third case using the upstream diamond box was already shown in Fig. 7. The fourth case consists of the entire computational volume, i.e., V_β is the overall mean velocity in the domain.
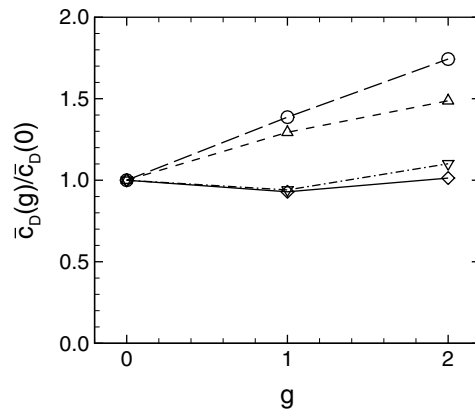
Fig. 14. Ratio of time-mean model coefficient using g-generation geometry $\bar{c}_D(g)$ to its $g = 0$ value, as a function of g. Long dashed line/circles: bounding box velocity; short dashed line/triangles: upstream half bounding box velocity; dash-dot line/inverted triangles: mean inflow velocity; solid line/diamonds: upstream diamond box velocity.

References

[1] R.H. Shaw, E.G. Patton, Canopy element influences on resolved- and subgrid-scale energy within a large-eddy simulation, Agr. Forest Meteorol. 115 (2003) 5–17.
[2] J.A. Kaandorp, E.A. Koopman, P.M.A. Sloot, R.P.M. Bak, M.J.A. Vermeij, L.E.H. Lampmann, Simulation and analysis of flow patterns around the scleractinian coral Madracis mirabilis (Duchassaing and Michelotti), Phil. Trans. R. Soc. Lond. B 358 (2003) 1551–1557.
[3] S. Chang, G. Iaccarino, C. Elkins, J. Eaton, S. Monismith, A Computational and Experimental Investigation of Flow Inside Branched Coral Colony, in: Center for Turbulence Research Ann. Res. Briefs, NASA Ames/Stanford University, 2004, pp. 45–54.
[4] C. van Ertbruggen, C.H.M. Paiva, Anatomically based three-dimensional model of airways to simulate flow and particle transport using computational fluid dynamics, J. Appl. Physiol. 98 (2005) 970–980.
[5] B. Mauroy, M. Filoche, E.R. Weibel, B. Sapoval, An optimal bronchial tree may be dangerous, Nature 427 (2004) 633–636.
[6] G.B. West, J.H. Brown, B.J. Enquist, A general model for the origin of allometric scaling laws in biology, Science 276 (1997) 122–126.
[7] B.B. Dykaar, P.K. Kitanidis, Determination of the effective hydraulic conductivity for heterogeneous porous media using a numerical spectral approach. 1. Method, Water Resour. Res. 28 (4) (1992) 1167–1178.
[8] B.B. Mandelbrot, The Fractal Geometry of Nature, W.H. Freeman, New York, 1982.
[9] C.J. Evertsz, B.B. Mandelbrot, Harmonic measure around a linearly self-similar tree, J. Phys. A: Math. Gen. 25 (1992) 1781–1797.
[10] K. Sarkar, C. Meneveau, Gradients of potential fields on rough surfaces: perturbative calculation of the singularity distribution function $f(\alpha)$ for small surface dimension, Phys. Rev. E 47 (2) (1993) 957–966.
[11] M. Brady, C. Pozrikidis, Diffusive transport across irregular and fractal walls, Proc. R. Soc. Lond. A 442 (1993) 571–583.
[12] M. Blyth, C. Pozrikidis, Heat conduction across irregular and fractal-like surfaces, Int. J. Heat Mass Transfer 46 (2003) 1329–1339.
[13] M. Blunt, Hydrodynamic force distribution on a fractal cluster, Phys. Rev. A 39 (11) (1989) 5801–5806.
[14] D.L. Johnson, J. Koplik, R. Dashen, Theory of dynamic permeability and tortuosity in fluid-saturated porous media, J. Fluid Mech. 176 (1987) 379–402.
[15] D.L. Koch, Attenuation of a compressional sound wave in the presence of a fractal boundary, Phys. Fluids 30 (10) (1987) 2922–2927.
[16] C. Pozrikidis, Unsteady viscous flow over irregular boundaries, J. Fluid Mech. 255 (1993) 11–34.
[17] P. Meakin, Z.-Y. Chen, J.M. Deutch, The translational friction coefficient and time dependent cluster size distribution of three dimensional cluster–cluster aggregation, J. Chem. Phys. 82 (8) (1985) 3786–3789.
[18] A. Adrover, M. Giona, Hydrodynamic properties of fractals: application of the lattice Boltzmann equation to transverse flow past an array of fractal objects, Int. J. Multiphase Flow 23 (1997) 25–35.
[19] D. Queiros-Conde, J.C. Vassilicos, Turbulent wakes of 3D fractal grids, in: J.C. Vassilicos (Ed.), Intermittency in Turbulent Flows, 2001, pp. 136–167.
[20] A. Staicu, B. Mazzi, J.C. Vassilicos, W. van de Water, Turbulent wakes of fractal objects, Phys. Rev. E 67 (2003) 066306.
[21] N.R. Wilson, R.H. Shaw, A higher-order closure model for canopy flow, J. Appl. Meteorol. 16 (1977) 1197–1205.
[22] M.R. Raupach, A.S. Thom, Turbulence in and above plant canopies, Annu. Rev. Fluid Mech. 13 (1981) 97–129.
[23] J. Finnigan, Turbulence in plant canopies, Annu. Rev. Fluid Mech. 32 (2000) 519–571.
[24] T. Maruyama, Optimization of roughness parameters for staggered arrayed cubic blocks using experimental-data, J. Wind Eng. Ind. Aerodyn. 46-7 (1993) 165–171.
[25] Y. Miyake, K. Tsujimoto, Y. Agata, A DNS of a turbulent flow in a rough-wall channel using roughness elements model, JSME Int. J. Ser. B: Fluids Thermal Eng. 43 (2) (2000) 233–243.

[26] E.G. Patton, R.H. Shaw, M.J. Judd, M.R. Raupach, Large-eddy simulation of windbreak flow, Bound. Layer Meteor. 87 (2) (1998) 275–306.
[27] J.D. Wilson, Representing drag on unresolved terrain as a distributed momentum sink, J. Atmos. Sci. 59 (9) (2002) 1629–1637.
[28] J. Cui, V.C. Patel, C.-L. Lin, Prediction of turbulent flow over rough surfaces using a force field in large eddy simulation, J. Fluids Eng. 125 (2003) 2–9.
[29] R. Verzicco, J. Mohd-Yusof, P. Orlandi, D. Haworth, Large eddy simulation in complex geometric configurations using boundary body forces, AIAA J. 38 (3) (2000) 427–433.
[30] K.G. Wilson, J. Kogut, Renormalization group and the ϵ expansion, Phys. Rep. 12C (2) (1974) 75–200.
[31] A. Yakhot, S. Orszag, V. Yakhot, M. Israeli, Renormalization group formulation of large-eddy simulations, J. Sci. Comput. 4 (1989) 139–158.
[32] L. Smith, S. Woodruff, Renormalization-group analysis of turbulence, Annu. Rev. Fluid Mech. 30 (1998) 275–310.
[33] M. Germano, U. Piomelli, P. Moin, W.H. Cabot, A dynamic subgrid-scale eddy viscosity model, Phys. Fluids A 3 (1991) 1760–1765.
[34] D.K. Lilly, A proposed modification of the Germano subgrid-scale closure method, Phys. Fluids A 4 (1992) 633–635.
[35] C. Meneveau, T.S. Lund, W.H. Cabot, A Lagrangian dynamic subgrid-scale model of turbulence, J. Fluid Mech. 319 (1996) 353–385.
[36] A. Scotti, C. Meneveau, Fractal model for coarse-grained nonlinear partial differential equations, Phys. Rev. Lett. 78 (5) (1997) 867–870.
[37] A. Scotti, C. Meneveau, A fractal model for large eddy simulation of turbulent flow, Physica D 127 (3–4) (1999) 198–232.
[38] M.F. Barnsley, Fractals Everywhere, second ed., Academic Press, London, 1993.
[39] D.A. Lyn, S. Einav, W. Rodi, J.-H. Park, A laser-Doppler velocimetry study of ensemble-averaged characteristics of the turbulent near wake of a square cylinder, J. Fluid Mech. 304 (1995) 285–319.
[40] J. Smagorinsky, General circulation experiments with the primitive equations: I. The basic experiment, Mon. Weather Rev. 91 (1963) 99–164.
[41] C. Meneveau, J. Katz, Scale-invariance and turbulence models for large-eddy simulation, Annu. Rev. Fluid Mech. 32 (2000) 1–32.
[42] P. Mason, D. Thomson, Stochastic backscatter in large-eddy simulations of boundary layers, J. Fluid Mech. 242 (1992) 51–78.
[43] F. Porté-Agel, C. Meneveau, M.B. Parlange, A scale-dependent dynamic model for large-eddy simulation: application to a neutral atmospheric boundary layer, J. Fluid Mech. 415 (2000) 261–284.
[44] E. Bou-Zeid, C. Meneveau, M. Parlange, A scale-dependent Lagrangian dynamic model for large eddy simulation of complex turbulent flows, Phys. Fluids 17 (2005) 025105.
[45] J.D. Albertson, M.B. Parlange, Surface length scales and shear stress: implications for land-atmosphere interaction over complex terrain, Water Resour. Res. 35 (7) (1999) 2121–2132.
[46] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (1972) 252–271.
[47] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35–60.
[48] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[49] Y.-H. Tseng, C. Meneveau, M.B. Parlange, Modeling flow around bluff bodies and predicting urban dispersion using large eddy simulation, Environ. Sci. Technol. 40 (2006) 2653–2662.
[50] J.H. Ferziger, M. Péric, Computational Methods for Fluid Dynamics, Springer, 1999.
[51] J.H. Ferziger, Numerical Methods for Engineering Application, Wiley, Interscience, 1998.
[52] F.P. Bertolotti, T. Herbert, P.R. Spalart, Linear and nonlinear stability of the Blasius boundary layer, J. Fluid Mech. 242 (1992) 441–474.
[53] P. Schlatter, N. Adams, L. Kleiser, A windowing method for periodic inflow/outflow boundary treatment of non-periodic flows, J. Comput. Phys. 206 (2005) 505–535.
[54] W.F. Lindsey, Drag of Cylinders of Simple Shapes, Technical Report 619, NACA, 1938.
[55] B.E. Lee, The effect of turbulence on the surface pressure field of a square prism, J. Fluid Mech. 69 (1975) 263–282.
[56] G.I. Barenblatt, Scaling, Self-similarity, and Intermediate Asymptotics, Cambridge University Press, Cambridge, 1996.
[57] F.M. White, Fluid Mechanics, McGraw-Hill, 1979.
[58] B.J. Vickery, Fluctuating lift and drag on a long cylinder of square cross-section in a smooth and in a turbulent stream, J. Fluid Mech. 25 (1966) 481–494.
[59] C.M. Cheng, P.C. Lu, R.H. Chen, Wind loads on square cylinder in homogeneous turbulent flows, J. Wind Eng. Ind. Aerodyn. 41 (1992) 739–749.
[60] W. Rodi, J. Ferziger, M. Breuer, M. Pourquie, Status of large eddy simulation: results of a workshop, J. Fluids Eng. Trans. ASME 119 (2) (1997) 248–262.
[61] A. Nakayama, S.N. Vengadesan, On the influence of numerical schemes and subgrid-stress models on large eddy simulation of turbulent flow past a square cylinder, Int. J. Numer. Meth. Fluids 38 (2002) 227–253.