A Physics Based Multi-Resolution Model for the Simulation of Turbulent Gases and Combustion

Daniel Barrero, Mathias Paulin, René Caubet
IRIT
118, Route de Narbonne, 31062 Toulouse, France
e-mail: {barrero|paulin|caubet}@irit.fr

Abstract: We present a technique for modeling the turbulent behavior of gaseous and combustion phenomena, based on the numerical approximation of the fluid's equations by using a seamless combination of different methods: a volumetric finite differences multi-resolution method, a wavelet model, a hierarchical model of turbulence, and a simplified flamelet model for combustion phenomena.

1 Introduction

One of the bigger problems found in computer graphics is the modeling of natural gaseous phenomena due to their complex and not clearly defined behavior and geometry. Although there had been taken several approaches to solve it, the main problem remains. Not only due to the restriction of the solutions to specific problems or subproblems, which by their specificity and heterogeneity are incompatible or very difficult to combine, but also because of the inherent complexity of the phenomenon.

From the physical point of view, these complex phenomena can be represented by means of the Navier-Stokes (N-S) equations and their extended forms[1,6,8,7,13,23,24,27]. In the case of turbulent combustion, there are some other specific models ranging from the simpler Arrhenius[19] approach to more complex ones like the EBU[4,19], PDF[28], or flamelet[16,21] models. And even then, the problem is still considered an open one due to the inherent complexity and interdisciplinary character of this phenomenon.

The principal problem when trying to solve the gas equations using traditional CFD techniques, is the high calculation time and resources required, and the difficulty in obtaining a specific effect by tweaking only initial boundary conditions. Only now, with the evolution of computer capacity, it is possible to use simplified CFD methods to obtain a rough approximation of these equations in reasonable time to be used in computer imagery as shown by Foster[13,1415]. Still, one of the principal problems is the calculation times and machine resources needed for a high quality animation, and the important error margin of the solution obtained for the lower scales of turbulence.

In the area of computer graphics, the problem has also been apprehended according to many different points of view. Particle systems[11] being the most popular to date, they are very efficient but have some drawbacks: the biggest one is that a real gas is a continuous medium and selecting only regions of it can cause unpredictable unrealistic results without a lot of hand-tweaking, due to the missing model of a gas mixing with its surroundings. To better reflect the rotational motion of gas turbulence some modifications have been proposed using point-vortex particles[9,29], but they require higher calculation times or are limited to 2D. Another popular approach is the utilization of stochastic methods in combination with particle systems[22,24,25], or the utilization of a randomly perturbed vector fields in combination with deterministic fields as proposed by Stam[24,25,26]. There are also pure geometric and heuristic models[10,22], where the approximation of the phenomenon depends directly on the artist ability to perturbate a geometric object, or on the utilization of noise functions[10] in combination with hypertextures[10] as proposed by Perlin[10].

Instead of using only one specific method to solve the problem, we try to solve it by means of a combination of different models. In this paper, we are going to describe our model which combines of a simplified form of a fast volumetric finite differences multi-resolution method, a wavelet model, a hierarchical model of turbulence, and a simplified flamelet model for combustion phenomena.

2 Physical Description of a Gas Behavior

To obtain a realistic representation of turbulence phenomena in gases, we must be able to understand how a gas is represented from the physical point of view. In general a gas behavior can be described by different factors: The general flow of the gas volume (laminar flow), turbulent flow, thermal buoyancy, convection and drag phenomena, gas diffusion, molecular dissipation, etc. And in the specific case of combustion, not only the thermodynamic processes produced by the combustion, but also the effects of combustion on the fuel source, the propagation of the flame and state changes of the burning gas. The way all these factors interact can be explained by the governing differential equations of a gas. Here we are going to use a simplified set of equations from the typical fluid dynamics (N-S) and combustion models.

2.1 Convection and Drag

The different factors that can influence the behavior of a gas will entail modifications in the speed ($\mathbf{u}$), direction, pressure (p), density (ρf), and temperature (T) of the fluid in the space-time. In general, these modifications can be expressed by the Navier-Stokes differential equations for the velocity of a gas:

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho f}\nabla p + v\nabla^2\mathbf{u} + \mathbf{F} \tag{1}$$

With: F=external forces applied to the fluid, ∇=gradient, v=kinematic viscosity directly related to the Reynolds number (Re) of the gas. If the velocities of the fluid are much smaller than the speed of sound, the fluid can be said to be incompressible:

$$\nabla \cdot \mathbf{u} = 0\,, \tag{2}$$

This equation (2) models how the velocity of a gas changes depending on convection ($(\mathbf{u}\cdot\nabla)\mathbf{u}$), its pressure gradient (∇p), and drag ($v\nabla^2\mathbf{u}$). The equation (1) models the convective and rotational velocity of the gas.

2.2 Temperature Evolution and Thermal Buoyancy

The effects of convection, diffusion and turbulence of the gas produced by the convection and diffusion of heat can be described by the Navier-Stokes system of equations for heat as follows:

$$\frac{\partial T}{\partial t} = \lambda\nabla \cdot (\nabla T) - \nabla \cdot T\mathbf{u} \tag{3}$$

Where λ represents turbulent and molecular diffusion, $\nabla\cdot T\mathbf{u}$=changes due to convection, and $\lambda\nabla\cdot(\nabla T)$=temperature changes due to diffusion and turbulent mixing.

Solving the system of equations (1)(2)(3), allows us to obtain an accurate model of movement for a hot turbulent gas. But there is still something missing, and that is the natural tendency of a hot gas to rise, and hotter regions of the gas tend to rise more quickly than the cooler ones that have mixed with the fresh air. This phenomenon which can modify significantly the motion of a gas is called thermal buoyancy, and it can be represented by defining a buoyant force in a gaseous element as follows[13]:

$$\mathbf{F} = -\beta\mathbf{g}(T_0 - T_k)\,, \tag{4}$$

With: **g**=gravity, β=thermal expansion coefficient, T_0=initial reference temperature, and T_k =average temperature on a specific gaseous region. We can apply this equation (4) directly in the equation (1) to obtain, in combination with equations (2) and (3), a really complete description of the behavior of a hot turbulent gas.

2.3 Modeling Turbulent Combustion

Modeling turbulent combustion is one of the most complex problems in the area of fluid dynamics, because of the interdisciplinary character of the phenomenon. Basically the complexity of turbulent combustion comes from three points:

1. Combustion without turbulence is a complex process involving many chemical reactions in very short times,
2. Turbulence is one of the most complex mechanisms in fluid dynamics,
3. Turbulent combustion is the result of the interaction of combustion with a turbulent flow.

As of now, in our model we have covered the point 2, which is the modeling of turbulence. Now, for the model of combustion in the area of computer graphics, there had been some simplified models for the spread of fire or combustion[9,25]. All these models assume that turbulence doesn't play any role in turbulent combustion, and the more sophisticated ones[25] are based on the Arrhenius[25] approach which can be probed false from a mathematical and thermodynamical point of view[19].

We had chosen to use the coherent flame model (CFM)[21] based on the flamelet model (Fig. 1), not only because it is one of the most accurate, but also because it integrates nicely with our turbulence model. The basic flamelet model assumes that the flame front is a continuous sheet separating fresh and burnt gases: it allows us to define the behavior of a turbulent combustion phenomenon but it needs to be complemented with some evaluation method of the flame surface.

The main advantage of the CFM model over the basic flamelet model is that it is based on a simple and intuitive description of the turbulent flame and that it allows us some complex features to be taken into account without additional complexity. Instead of trying to define turbulent combustion in terms of one-point quantities like the crossing frequencies of flamelets, it considers the flame surface as the important quantity controlling the reacting flow and uses it as the parameter to model. This means that it is possible to define the turbulent combustion behavior as by the following set of equations:

$$\dot{\bar{w}} = w_L \, \Sigma \tag{5}$$

$$\frac{\partial \quad \bar{\rho}}{\partial \quad t} + \frac{\partial \bar{\rho} \tilde{\mathbf{u}}_i}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\frac{\mu_t}{\sigma_\Sigma} \frac{\partial \Sigma}{\partial x_i} \right) + \bar{\rho} E_s \, \Sigma - w_L \frac{\Sigma^2}{\bar{Y}} \tag{6}$$

2.4 Hierarchical decomposition of turbulence

In fluid dynamics, it has been observed that a turbulent movement field can be decomposed in many levels of turbulence, from the biggest scales corresponding to the global fluid flow (laminar), and the lower ones representing turbulence phenomena. Moreover, when looking very closely to turbulent phenomena, it is possible to identify very clearly a set of forms that seems to have a tendency to appear in a repetitive and uncorrelated way, i.e., vortexes are clearly identifiable in almost any turbulent flow. Even more astonishing is the fact that if we take a closer look to the geometry of other less evident turbulence phenomena, it is possible to see that its geometry can be decomposed in a combination of simpler forms of turbulence (Fig.4).

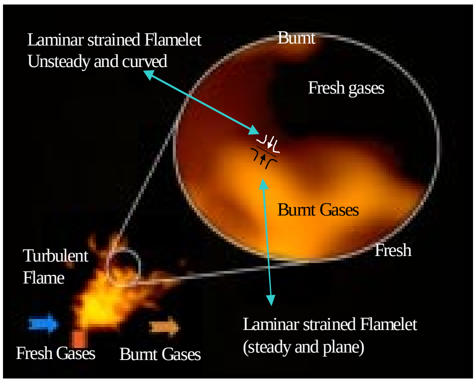

Fig. 1. The flamelet regime

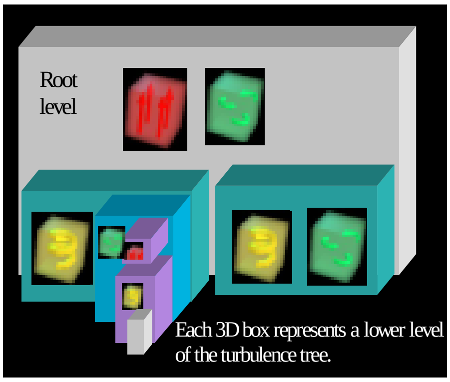

Fig. 2. Turbulent movement fields hierarchy.

This observation has given birth to a different set of models in the area of CFD that take advantage of this situation to represent turbulence. This decomposition of turbulence also helps us to understand the evolution of energy within this complex system by means of the Kolmogorov cascade of energy, which models the energy transfer between the different turbulence levels. Thus it can be used to control the turbulence behavior of a gas.

In general this hierarchical definition of turbulence can be expressed by the following equation, for the case of the velocity of the gas:

$$\mathbf{u}(\mathbf{x},t) = \alpha \boldsymbol{U}_g(\mathbf{x},t) + \beta \boldsymbol{U}_p(\mathbf{x},t)\,, \qquad (7)$$

Here we have a complex movement field $\mathbf{u}(\mathbf{x},\mathrm{t})$, that is defined by the composition of two sets of fields: big scale $\boldsymbol{U}_\mathrm{g}(\mathbf{x},\mathrm{t})=\Sigma\lambda_\mathrm{i}\mathbf{u}_\mathrm{gi}(\mathbf{x},\mathrm{t})$ where $\mathbf{u}_\mathrm{gi}(\mathbf{x},\mathrm{t})$=field function, and small scale $\boldsymbol{U}_\mathrm{p}(\mathbf{x},\mathrm{t})=\Sigma\gamma_\mathrm{i}\mathbf{u}_\mathrm{si}(\mathbf{x},\mathrm{t})$, where $\mathbf{u}_\mathrm{si}(\mathbf{x},\mathrm{t})=\mathbf{u}(\mathbf{x},\mathrm{t})=(7)$, which corresponds to a set of another turbulent fields, giving us a recursive definition of turbulence.

From this equation (7) it is possible to infer that a turbulent field can be represented by a n-ary tree of fields, with a depth of m (Fig. 2). The recursive definition of this function ends when, for every u_i of a specific level, the U_pi are equivalent to an empty set. That is it is defined only by the totality of the $\boldsymbol{U}_\mathrm{gi}$.

This hierarchical representation of turbulence also allows the utilization of multi-resolution numerical methods to approximate the solution of the N-S equations.

2.4.1 Autosimilar form of the Navier-Stokes equations

Based on these observations, there has been a lot of research to produce an autosimilar form for the Navier-Stokes equations. This has been finally somewhat possible with the help of wavelets under certain conditions[10], namely that the initial value of vorticity $w_0(\mathrm{x})$ has to be small in a measure space of type Morrey-Campanato[1,6] and some special homogeneity conditions. This means that the unity of the solution can be guaranteed if these conditions are satisfied. It can also be probed that it is possible to approximate the solution to N-S, using a wavelet analysis of frequency and scale without having a completely adapted space, but this is out of the scope of this article and a good discussion of it can be found on [1] and [6]. With this model, the equation (1) for the velocity field becomes:

$$\gamma_u(r) = \mathrm{E}(\mathbf{u}(r_0 + r)\mathbf{u}(r_0)) \sim \varepsilon^{-2/3} r^{2/3}\,, \qquad (8)$$

which is the dual expression of the spectral density for the power of the velocity field dependence in $\varepsilon^{-2/3}r^{2/3}$. These are the successive associations between the speed increase measured in a scale *r*, the auto-correlation of the speed signals from one side, and the spectrum of auto-correlation from the other side. This behavior suggests the auto-similarity properties of the N-S equations. In effect, the equation (1) stays the same for the following scale changes:

$$r \longrightarrow \lambda r,\ t \longrightarrow \lambda^{1-H_t},\ \mathbf{u} \longrightarrow \lambda^{H_\mathbf{u}} \tag{9}$$

The time-space variables are dynamically linked : the scale change of the second is subordinated to the first, and so on. The velocity field presents as a result the following autosimilar relationship:

$$\mathbf{u}(\lambda r, \lambda^{1-H_t}) \longrightarrow \lambda^{H}\mathbf{u}(r,t) \tag{10}$$

which is pretty similar to the one for the fBm, but the analogy with the fBm motion is not complete because the velocity field is not a gaussian process.

The main interest of this approach is the possibility of using it to obtain a solution of the 3D Navier-Stokes equations for a homogeneous region of the space. As a consequence, if we use a finite differences method, we can apply the wavelet approach in the subspace defined by each cell in order to obtain a solution for the scales of turbulence lower than the minimum cell size.

3 Building a useful model of a gas for computer graphics

3.1 Modeling the simulation environment

The method we propose to calculate the effect of an object over a fluid flow is a simplification of the traditional finite volume solution of the Navier-Stokes equations[7,13]. Like the previous work of Foster[7], we approximate the scene by a series of regular voxels axially aligned to a coordinate system x,y,z. And likewise if a portion of the medium (gas) surrounding the objects is voxelized using the same coordinate system, then the boundaries of the objects can be made to coincide with the faces voxels. The resulting grid can be used to solve physics-based equations in a straightforward way. The main difference is the utilization of a multi-resolution grid based on an octree-like approach (Fig. 3).

For each cell on the grid, we define the variables for the average temperature $T_{i,j,k}$, pressure *p*, and fuel quantity *C* (which can be defined for each object via a fuel map), within the cell. Likewise in the center of each face of a cell, we define the gas velocity perpendicular to that face: this leads to the velocities shown in the Fig. 3.

3.1.1 Turbulence Basis

As it has been explained before it is possible to identify a set of forms that seems to have a tendency to appear in a repetitive and uncorrelated way in turbulence phenomena. Based on this observation, we propose to simplify the modeling of turbulence phenomena by defining a turbulence basis as a mathematical group using a set of linearly independent movement fields. That, when combined will be able to reproduce complex turbulence phenomena[1,2,3,6] (Fig. 4). Each field corresponds to a geometrical form observed in the real phenomenon, or a physically accurate turbulence model evaluated into a parametric space (i.e. wavelets or finite differences)

The main interest of this approach is the possibility of applying it directly within a hierarchical model of turbulence, and the ability to allow a seamless combination of different models of turbulence for different scales or types of turbulence. Also it is easier to obtain specific special effects by using deterministic fields to modify the fluid’s flow.

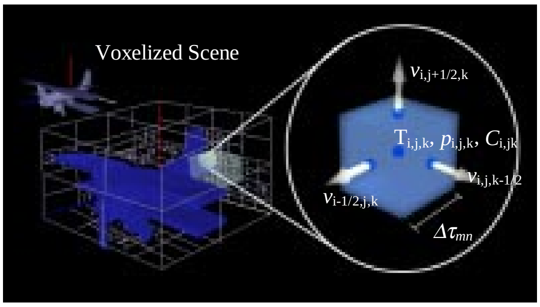

Fig. 3. Scene represented by a multi-resolution grid

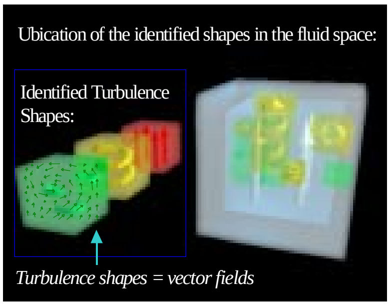

Fig. 4. Turbulence basis concept

3.1.2 Turbulence Hierarchy

We use the straight definition of a turbulence hierarchy as explained before, to build our model of turbulence as a n-ary tree of fields, where each level of the three contains a set of large scale fields and a set of lower turbulence scale fields which are in the next level of turbulence (Fig. 2 and Fig. 5).

The set of large scale fields in a level is a combination of a set of deterministic fields from the turbulence basis at that turbulence size, and the corresponding level of the multi-resolution grid. This is possible by the consideration of each of the voxels of the grid corresponding to this level as a special large scale turbulence field, meaning that it can contain lower scales of turbulence in it, following the hierarchical definition of turbulence Furthermore, these lower levels are not necessarily of the same type. This allows us to combine the finite elements and wavelet methods in a seamless manner. All this can be represented more clearly in a graphical way as shown by the Fig. 5.

3.1.2.1 Applying the equations to the grid

To solve the system of equations (1),(2),(3),(5) and (6), we have to rewrite them in an appropriate form that is applicable to the voxel grid, using a numerical method known as finite differences. The main idea is that a differential term such as $\partial T/\partial y$, can be approximated using Taylor series, giving us a model directly applicable to a regular voxel grid once rewritten in terms of the free variables of the grid as follows:

$$\begin{aligned}T^{n+1}_{i,j,k} = T^{n}_{i,j,k} + \Delta t\{(1/\Delta\tau)[(T\mathbf{u})^{n}_{i-1/2,j,k} - (T\mathbf{u})^{n}_{i+1/2,j,k} + (T\mathbf{v})^{n}_{i,j-1/2,k} - (T\mathbf{v})^{n}_{i,j+1/2,k}\\ + (T\mathbf{w})^{n}_{i,j,k-1/2} - (T\mathbf{w})^{n}_{i,j,k+1/2}]\\ + \frac{\lambda}{\Delta\tau^2}[(T^{n}_{i+1,j,k} - 2T^{n}_{i,j,k} + T^{n}_{i-1,j,k}) + (T^{n}_{i,j+1,k} - 2T^{n}_{i,j,k} + T^{n}_{i,j-1,k}) + (T^{n}_{i,j,k+1} - 2T^{n}_{i,j,k} + T^{n}_{i,j,k-1})]\}\end{aligned} \tag{11}$$

A good discussion of the method to rewrite the equations can be found on[13,14,15], and is not going to be repeated here. All the equations of the system (1),(2),(3)(5) and (6) can be converted in a similar form.

The basic idea of the method of finite differences in a regular grid is to obtain a solution of the equations for each grid cell, in function of the actual state of the cell and the surrounding ones. This method can be easily adapted to a multi-resolution grid by applying the method for regular grids to each one of the sublevels of the grid (which are themselves regular and correspond to a full level of the turbulence hierarchy) from the biggest level to the lower resolution ones, using the obtained intermediate results to set the initial boundary conditions of the next level of turbulence. This propagation can be easily accomplished by using the turbulence tree.

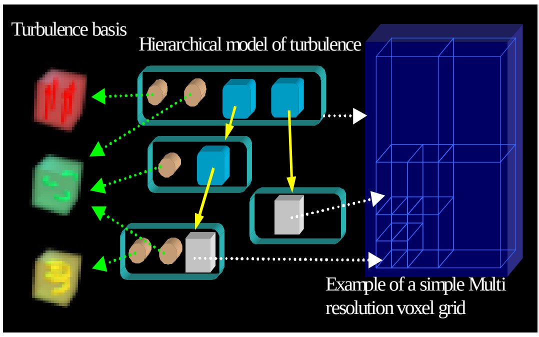

Fig. 5. Relationship between the different parts of the turbulence model

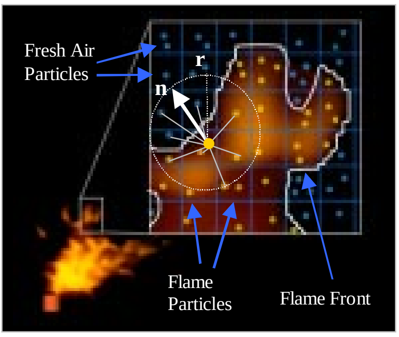

Fig. 6. Following the flame front

3.1.2.2 Ensuring Accuracy and Stability

The approximation of the simulation environment as a voxel grid is one of the main sources of efficiency for our algorithm, which is improved by the utilization of the turbulence hierarchy, but it has a drawback. The problem is that a low resolution variable sampling can introduce error into the calculation. Because the free variables $\mathbf{u}$ and T are calculated at fixed positions of the space, an error is introduced into T and $\mathbf{u}$ when the finite difference is calculated. This means that mass is created or destroyed as a side effect of the algorithm. This is very evident in a regular grid method and can be partially resolved by our utilization of a multi-resolution grid.

Anyway, it is necessary to correct this change in mass to ensure the accuracy of the simulation. To correct this change in mass, we need to ensure that for any point in the scene (unless we want this to happen), the mass flowing in is the same mass flowing out. This situation is characterized by a constraint equation that is part of the Navier-Stokes equations as expressed in equation (2).

For a single cell, the left hand side is approximated using the Taylor series method, and rewritten in term of the grid variables: This is the same approach used by Foster[13,14,15] to obtain a divergence field to account for this phenomenon.

We use this approach only if the divergence field error is too large, and only after we have calculated the effects of lower scale turbulence (lower than the grid size) using the wavelet approach. The main reason for that is that in general the calculation of the lower scales of turbulence diminishes the error by a big factor, making almost unnecessary the application of the relaxation adjustment.

3.2 Following the flame front

The most complex problem for the modeling of turbulent combustion phenomena is the tracking of the flame front. This problem is very similar to the tracking the surface of a fluid. In fact, so similar that, we use an approach based on the one proposed by Foster[14,15] for tracking a fluid surface with minimal modifications.

This approach consists in identifying each cell as a gas, free, or static cell (solid object), and in keeping series of tracking particles at each of the cells containing a gas or in the surface. Each of these particles will be updated according to its relative position to the flame front. Initially, all particles are going to be marked as fresh air particles and are going to be marked flame particles following the resolution of the combustion model for that particle position in that cell. To find the flame front, we must analyze each of the particles on a gas cell adjacent to one or more free cells in comparison to the particles on the free cells (Fig. 6), to obtain a implicit representation of the flame front's surface. This approach will simplify the application of the flamelet model and also suggests a direct method for visualizing the flame surface.

3.3 Controlling the evolution of the model

One of the main advantage of this model is that it exists many ways in controlling the model, according to the final desired result. In general, the evolution of the hierarchical model is given by Kolmogorov energy cascade, which determines the creation or apparition of new fields in the hierarchy or the dynamic subdivision of the grid to take into account high energy changes. Modifying the energy cascade allows for a great deal of control on the observed behavior of a gas, especially on the type of turbulence observed.

One way of controlling the gas behavior is to use boundary conditions applied directly on the grid. This is also the method used to model interaction with solid objects, i.e. a solid object doesn't allow a gas to pass through it but it allows it to freely flow following a path tangential to its surface. In this case, we will set the velocity of voxels corresponding to the object to zero and the adjacent ones to a velocity of zero in the direction of the object and some initial velocities (depends on the object material, roughness, etc.) tangential to the object face.

One way to obtain a specific effect is to force the gas to follow some fixed paths, or to make appear a specific form of turbulence by using deterministic functions.

No matter what method of control is used the initial boundary conditions of the simulation and the parameters of the energy cascade are always present. This way, it is possible to assure that the resulting animation will be a natural looking one, by the underlying resolution of the Navier-Stokes equations.

4 The turbulent gas behavior algorithm

The algorithm for calculating the behavior of hot turbulent gases and combustion is composed of two stages. Given the physics-based nature of the method, it resembles the first preparation steps in a fluid dynamics program, but simpler in many ways.

4.1 Preparation Stage

The preparation work that an animator must do can be divided in seven steps. Some of them are made first in an automatic way, and can be fine hand tuned in a later step:

1. Set the subdivision range to decompose the environment in a coarse voxel grid, with a biggest size length between $\Delta\tau$ and $\Delta\tau_m$, which will be used to subdivide the solid objects.
2. Set the different fluids space of influence within the simulation environment.
3. Set the subdivision parameters of the environment into a coarse voxel grid to better suit the animator's requirements. An initial automatic subdivision is done based on the objects and fluids positions using the size length limitations.
4. Set boundary conditions for velocity and temperature in the grid, that is, place all the heat sources and sinks, combustion sources, fuel cells, and assign all the fuel maps to the objects
5. Create all the deterministic fields to control the general fluid flow.
6. For each different fluid in the simulation:
   - 6.1. Consider the different fluids physical properties: viscosity, thermal expansion, molecular diffusion, Reynolds number, etc.
   - 6.2. Set the different fluids maximum space of influence and place them in the environment
7. Determine Δt from the minimum 1/30 of a second and the largest stable time step.

4.2 Automatic Simulation Stage

The second stage corresponds to the automatic part of the process, namely the simulation step that can be described by the pseudo-algorithm shown in Fig. 7.

```
For each time frame do
  Apply boundary conditions to the objects
  Update Fuel map/cells
  Actualize boundary conditions that rely on the fuel maps
  Generate flame front tracking particles
  Push Energy: Top down energy redistribution
  For each level of the hierarchy
     For each field in the level
          If is deterministic
                Update field
                Transfer energy / heat between overlapping fields
          Elsif is voxel cell
                Update cell variables using finite differences
                If not subdivided
                   Update Wavelet conditions
                Fi
                Find divergence field for the gas to conserve mass
                Apply relaxation adjustment if necessary
          Fi
      End
   End
   Pull Energy: Down to top free energy redistribution
   Update flame front tracking particles (move and advect)
End
```

Fig. 7. Pseudo-algorithm for the automatic simulation stage

5 Rendering

Many approaches for the rendering of gaseous phenomena have been presented in recent years and are not going to be explained here. Basically, we use two methods:

To help the animator to set up the simulation environment and see the output in real time, we use a system of mass-less particles which are advected by the velocity field produced by the simulation. The position of a particle in a given time step can be obtained by using the gas velocity at the particle position $\mathbf{u}(\mathbf{x},t)$ on the equation:

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \Delta t\mathbf{u}(\mathbf{x},t) \qquad (13)$$

To produce a high quality output we use an extended version of Stam's[24,26] model of fuzzy blobs[24,26]. The main addition is the utilization of a hierarchy of fuzzy blobs[3], highly coupled with our turbulence model. This model is more oriented to the visualization of hot turbulent gases for that reason we are currently working on developing a method more appropriate to render combustion phenomena (fire).

Table 1: Some Control variables used for the sample images

Image	Max Tree Depth	New Childs	Basis fields	Src T	Air T	Max. Blobs	Grid Size
1-L / R	2 / 10	3 / 3	3 / 3	50^0C	24^0C	700/700	8 / 8
2	15	2	2	50^0C	24^0C	800	8
3	12	2	2	50^0C	24^0C	800	8
4	12	3	2	30^0C	20^0C	900	22
5	14	2	2	50^0C	20^0C	800	12

6 Conclusion Remarks and Future Directions

The main advantage of this turbulence model is the seamless combination of different modeling methods. Which simplifies controllability of the simulation by the combination of physical parameters and the superposition of deterministic fields for extended animator control, without sacrificing the accuracy of the simulation.

Also as the heat transfer between gases and solid objects is being taken into account it opens a great deal of possible effects to be developed in the future.

A problem with this method is that the grid cell orientation can affect the results, although it is partly solved by the way we evaluate the lower scales of turbulence within each cell. This can be improved by using non-structured unaligned grids.

As a concluding remark we can say that there is still a lot of work left to do because of the complexity of this type of phenomena.

References

1. Abry P., “Ondelettes et Turbulences”, Colection Nouveaux Essais, Diderot Editeur, Arts & Sciences, Paris, France, 1997.
2. Barrero D., Hernández J.T., “Modelo para la Simulación de Fenomenos Turbulentos para Ambientes de Realidad Virtual”, CLEI 1996, Bogota, Colombia, 1996.
3. Barrero D., DaDalto L, Paulin M., Caubet R, “Modélisation des Phénomènes Turbulentes dans les Milieux Participantes”, V^{eme} Journées AFIG, Rennes, France, December 1997.
4. Bray K.N.C, Libby P.A., “Topics in Applied Physics”, Vol 44, p.115, Springer-Verlag, 1980.
5. Brodlie K.W. et al., “Review of Visualization Systems”, Advisory Group on Computer Graphics Technical Report, Loughborough University of Technology:Loughborough, Leicestershire, 1995.
6. Cannone M, “Ondelettes, Paraproduits et Navier-Stokes”, Colection Nouveaux Essais, Diderot Editeur, Arts & Sciences, Paris, France, 1995.
7. Computed Fluid Dynamics Journal, IEEE Press, Jan.-Dec. 1994, Jan.-April. 1995.
8. Chen J.X. et al. “Real-Time Fluid Simulation in a Dynamic Virtual Environment”, IEEE Computer Graphics and Applications, pp. 52-61, May-June 1997.
9. Chiba N., Ohkawa S, Muraoka K, Miura M, “Two-dimensional Simulation of Flames, Smoke and the Spread of Fire”, J. of Vis. And Comp. Animation, 5,pp.37-54, 1994.
10. Ebert D.S., Musgrave F.K., Peachey D., Perlin K., Worley S., "Texturing and Modeling a procedural approach", Academic Press Inc., Cambridge MA, 1994.
11. Ebert D.S., Carlson W.E., Parent R.E., "Solid Spaces and Inverse Particle Systems for controlling the Animation of Gases and Fluids", The Visual Comp.,10, 1994.
12. Foley J. et al. “Computer Graphics Principle and Practice”, USA, Addison Wesley, 1992.
13. Foster N., Metaxas D., "Modeling the Motion of a Hot, Turbulent Gas" , ACM Computer Graphics, SIGGRAPH 97, Addison Wesley, August 1997.
14. Foster N., Metaxas D., "Realistic Animation of Liquids", Graphics Models and Image Proc., 58(5), pp 471-483, 1996.
15. Foster N., Metaxas D., "Controlling Fluid Animation", Proceedings of CGI’97, 1997.
16. Marble F.E, Broadwell J., "The coherent flame model for turbulent chemical reactions", Project SQUID, Report TRW-9-PU, 1977.
17. Karamcheti K, "Principles of Ideal Fluid Aerodynamics", 2nd edition, Kreiger, 1980
18. Kass M. , .Miller G., “Rapid, Stable Fluid Dynamics for Computer Graphics”, ACM Computer Graphics, SIGGRAPH 90, pp.19-57, August 1990.
19. Kuo K.C. , “Principles of Combustion”, John Wiley Intersci., 1986
20. Post F.H., van Walsum T., “Fluid Flow Visualization, in Focus on Scientific Visualization”, Springer-Verlag, 1993.
21. Pope S, Cheng W, “The stochastic flamelet model of turbulent premixed combustion", Twenty Second Symposium on Combustion, p. 781, The Combustion Institute, 1988.
22. Reeves et al. “Approximate and Probabilistic Algorithms for Shading and Rendering Particle Systems”, ACM Computer Graphics, SIGGRAPH 85, pp.313-322, July 1985.
23. Rogallo R.S., Moin P., “Numerical Simulation of Turbulent Flows”, Annual Review of Fluid Mechanics, pp. 99-137, 1984.
24. Stam J., Fiume E., “Turbulent Wind Fields for Gaseous Phenomena”, ACM Computer Graphics, SIGGRAPH 93, pp 369-373, Addison Wesley, August 1993.
25. Stam J., Fiume E., “Depicting Fire and Other Gaseous Phenomena Using Diffusion Processes”, ACM Computer Graphics, SIGGRAPH 95, pp.129-136, Addison Wesley, August 1995.
26. Stam J., "Multi-Scale Stochastic Modeling of Complex Natural Phenomena". Ph.D. Thesis, Dept. Of Computer Science, University of Toronto, 1995.
27. Streeter V.L., “Mecanica de los Fluidos”, Colombia, McGraw Hill, Octava Edición, 1995.
28. Williams F.A., “Combustion Theory”, 2nd ed., Benjamin Cummings, Menlo Park, 1985.
29. Yaeger L et al. “Combining Physical and Visual Simulation - Creation of the Plane Jupiter for the film 2010”, ACM Computer Graphics, SIGGRAPH 86, pp.85-93, Addison Wesley, 1986.

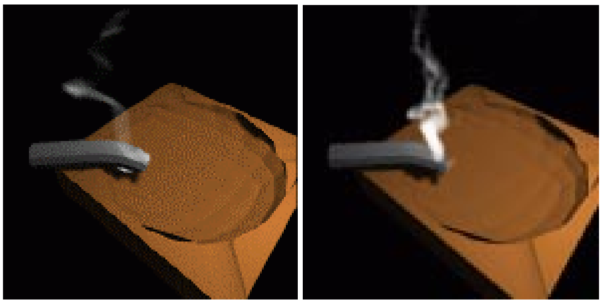

Image 1. Smoke with different hierarchy levels, laminar flow (left), turbulent flow (right). Shows the direct relationship between the turbulence quantity and the depth of the turbulence tree when using mainly the turbulence basis for the simulation.

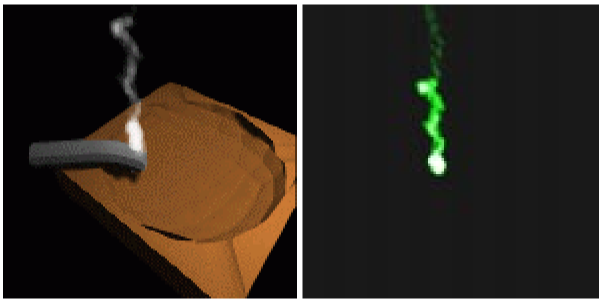

Image 2. Smoke in the visible an infrared spectrum (smoke only in the infrared for clearness). In the infrared the smoke appears to be denser than in the visible spectra due to the fact that the difference of temperature of some particles with their environment is more important than their density.

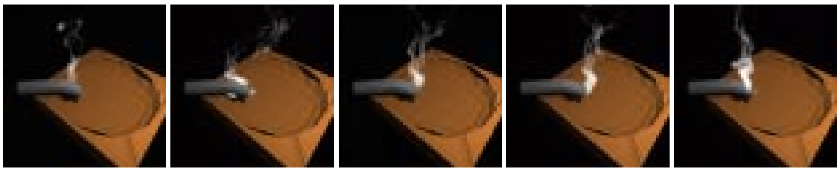

Image 3. Some images of an animation sequence

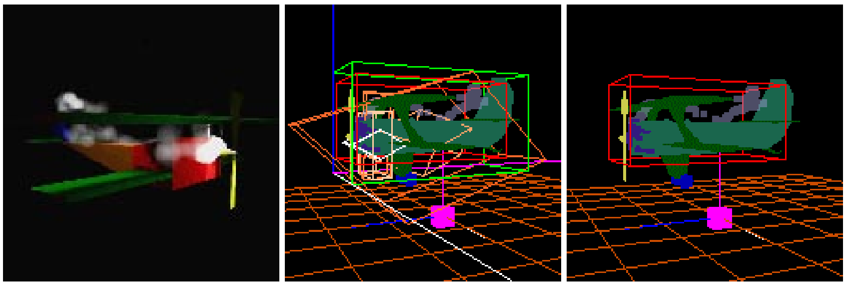

Image 4. Interaction of a gas and a complex object. The smoke leaving the engine is influenced by the wind field produced by the propeller. It is possible to observe a division of the smoke caused by the interaction with the wings and fuselage. Center image: boundaries for the highest levels of the hierarchy. Right image: real time smoke rendering as a particle system.

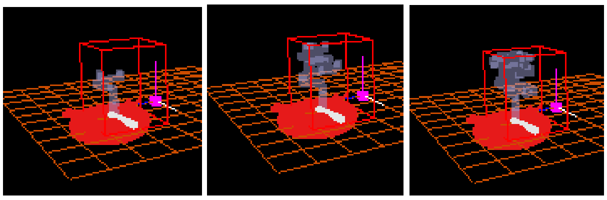

Image 5. Similar test scene as image 1 using the real time OpenGL viewer, this time it uses a better combination of the multi-resolution grid and the turbulence basis.