Xie *et al. EURASIP Journal on Wireless Communications and Networking* (2019) 2019:252
https://doi.org/10.1186/s13638-019-1584-0

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

A simulation method of three-dimensional cloud over WRF big data

Yonghua Xie*, Xiaoyong Kou and Ping Li

Abstract

Nowadays, due to the expansion of people's living ranges and the impact of human life on the natural environment, climate changes fiercely than before. In order to observe the changing climate environment accurately, multi-modal sensors are used to collect the various data around us, and we could analyze and predict the weather based on these collected data. One of the applications is 3D visualization simulation, and the 3D visualization simulation of cloud data has always been the research hotspot in the field of computer graphics and meteorology. Currently, it is a key challenge to resolve the problems of 3D cloud simulation, such as reducing complexity of modeling and computation and improving the real-time performance. Technically, a method for data modeling and optimizing based on Weather Research and Forecasting (WRF) is proposed in this paper, aiming to solve the problems of the existing 3D cloud simulation and realize 3D virtual simulation of real-world cloud data. According to the characteristics (e.g., color, size, shape) of the cloud, the spherical particle system is designed to model, and the initial color, size, shape, and other attributes are given to these spherical particles to realize the modeling of WRF cloud data. From the perspective of new particles' generation, the level of detail (LOD) technique, based on the relationship between the quantity of new generated spherical particles and the distance of the viewpoint, is used to change the quantity of new particles generated in real time according to the distance of the simulated scene distance. Finally, illumination model is introduced to render and simulate the modeling particles. Experimental simulation results verify the effectiveness of this method in improving the modeling and rendering speed of cloud data as well as the fidelity of the 3D virtualization model.

Keywords: Visualization, Cloud data, Multi-modal sensors, WRF, Particle system

1 Introduction

Recently, various sensors are developed to collect the data in different areas, and different sensors collect different types of data [1]. In order to collect the needed data including the temperature, pressure, and humidity, the sensors are distributed in the corresponding areas. In addition, the data collected by the sensors are widely used in the Weather Research and Forecasting (WRF). The collected data is analyzed, and the analyzed results are leveraged to forecast the trend of the weather [2].

It has been a hot and difficult point in the study of computer graphics that computer models and graphic algorithms are adopted to realize the simulation of natural objects with irregular shapes and abundant surface details, such as clouds, flowing water, atmosphere, rain, and snow [3].The complex visual and physical properties of clouds, as an integral part of weather systems, have attracted the attention of meteorologists and physicists and are of great importance for simulation. It is universally known that the physical state and physical meaning of different natural scenes are different. Thus, it is practicable to integrate the physical information into the process of simulation and rendering. The cloud data of WRF model have cloud physical variables, including pressure, air temperature, wind direction, and speed, which are concerned with the structure and characteristics of the cloud [4, 5]. However, it is difficult to describe it with accurate methods and models, and even more difficult to truly display the three-dimensional cloud scene, due to its complex structure, uncertain dynamic characteristics, and special lighting effects of clouds [6, 7]. From the perspective of graphics, the characteristics of cloud, such as no definite

*Correspondence: xyh_76@nuist.edu.cn
School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China

boundary, no definite surface, and complex illumination effect, make the traditional geometric modeling method not suitable for cloud modeling. Therefore, WRF cloud data simulation has always been one of the most challenging research directions in the field of meteorological application [8–10].

Clouds are mixture of ice crystals and water droplets. In order to vividly describe clouds' structure, particle system has been considered in that it is the most successful graphic generation algorithm to simulate the irregular fuzzy objects. Reeve was the first to put forward the concept of particle system and its mapping algorithm [11, 12]. Fuzzy objects are represented by plenty of particles, and each of them has a certain life cycle, and other properties, such as color, shape, size, and speed. Hence, the simulation effect depends on the number and size of particles. The smaller particles are and the larger the number, the more details the simulated objects have. Meanwhile, it takes longer time to run and consume more computing resources [13, 14].

On the one hand, the platform with better computing performance can be used to reduce the running time and computing resources [15, 16]. On the other hand, the generation algorithm of the particle system can be optimized to reduce the number of particles in the simulation system, which also has good simulation effect. In traditional geometric modeling, the level of detail (LOD) technique means that the number of vertices in the geometric model is determined according to the distance of the object from the viewpoint [17–19]. For particle system, object simulation can also be carried out from this idea, that is, when the viewpoint is relatively close, more particles can be generated each time; conversely, fewer particles can be generated when the viewpoint is relatively far. Therefore, the computing time is reduced and the rendering efficiency of the system is improved.

With the observation above, it is still a challenge to model the cloud based on weather forecast data which could reflect the structure and characteristics of cloud physically. Meanwhile, it is a time-consuming procedure of modeling based on weather forecast data, which makes it difficult to render clouds in short time [20]. Moreover, influenced by the illumination, some parts of cloud may be gloomy, but other parts may present bright effect. Hence, an illumination model is introduced to describe the cloud layer illumination in this paper.

The main contributions are listed as follows:

- Analyze the weather forecast data and build a spherical particle systematic model that constructs the basic shape of clouds.
- Analyze the relationship between the number of particles generated in real-time and the distance of the simulated scene distance as well as build an illumination model to render the cloud.
- Sufficient experimental evaluation and comparative analysis are carried out to verify the effectiveness and validity of the proposed method.

The rest of this paper is organized as follows. Section 2 describes the proposed method. In Section 3, experiments and evaluations are conducted. In Section 4, related work is presented. Section 5 concludes the paper and presents the future work.

2 Model description and methodology

In this section, we propose a particle system of cloud, which consists of the structure model, simplification model, and illumination model. Specifically, the shape of cloud is described by the structure model; the computing performance is improved by the simplification model. Light transport in clouds could be described by its illumination model, which conforms to the laws of radiative transfer.

2.1 Basic concepts

Particle system is a simulation method for irregular fuzzy object, in which objects are defined as thousands of irregular and randomly distributed particle sets, each particle has a certain life cycle, constantly moving and changing its shape every moment. In this way, the dynamic change of the shape and characteristics of the scene are described by the collection of many particles, not a single particle. The particle system fully embodies the motion and randomness of the irregular fuzzy object, which conforms to the physical nature of the object, and thus can well simulate natural landscapes such as fire, cloud, water, and forest [21]. The basic steps of object modeling with particle system are shown in Fig. 1:

2.2 Particle system of cloud

Each operation of the particle system is a process computing model, so it can be combined with any model that describes the motion and characteristics of the object. To truly simulate an object, particles are given the following properties: position, size, speed and direction of movement, color, transparency, and life cycle. The WRF model provides with various options for parameterizations of the boundary layer physics [22]. It provides the user various options for the parameterization of microphysics and cumulus parameterization, such as the composition, size, shape, and concentration of cloud [23].

The common method to simulate cloud in particle system is to set up a sphere and fill the interior of the sphere with particles. In order to ensure the computational efficiency [24], the size of the sphere is limited to a certain

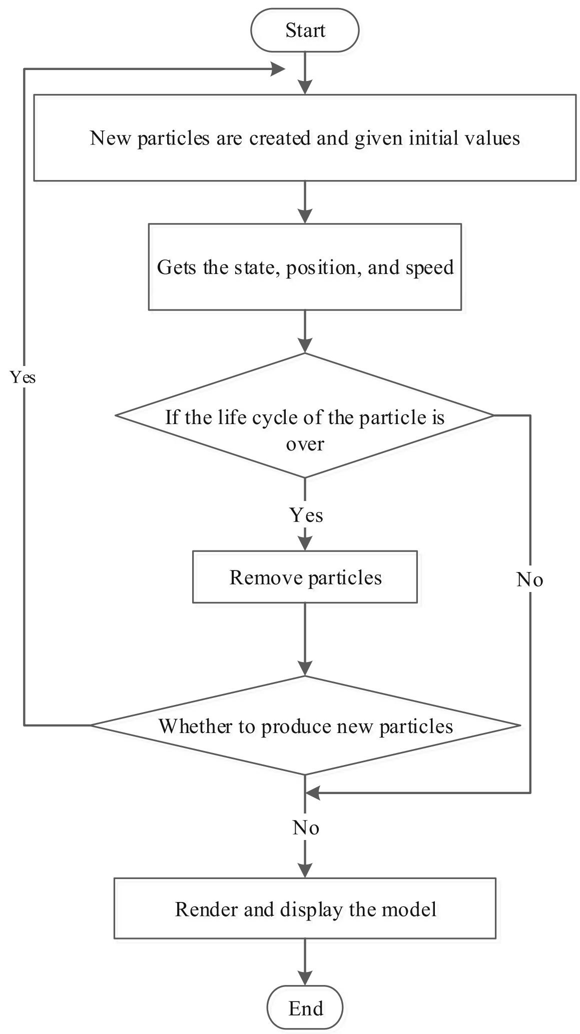

Fig. 1 Particle system flow chart of rendering

extent, and it cannot approach the horizontal distribution of clouds accurately. At the same time, the randomly generated particles are dense in the center of the sphere and sparse in the periphery, which satisfy the objective perception of human beings, but lose the real distribution law of cloud particles. Hence, it is vital to describe the real distribution law of cloud particles according to the WRF model. Among of the WRF data, cloud optical thickness is an important parameter to describe the optical properties of clouds [25]. Its definition is as follows:

$$\lambda = \Delta h \int \Gamma \pi r^2 n(r) dr, \tag{1}$$

where Γ represents extinction efficiency factor, and the cloud optical thickness λ represents a function of cloud droplet radius, wavelength, and refractive index. Δh represents the thickness of the clouds; $n(r)$ represents the number of particles with radius r per unit volume. In the visible band, when the scale of cloud particles increases to a certain value, Γ goes to 2, then

$$\lambda = 2\pi \Delta h \int r^2 n(r) dr. \tag{2}$$

Obviously, the cloud optical thickness is related to the droplet concentration per unit volume, droplet radius, and cloud thickness. It is shown that the appearance of clouds with large optical thickness is characterized by thick cloud cover and high concentration of cloud droplet particles. Therefore, the optical thickness of the cloud can be used to approximately represent the thickness of the cloud, which is in line with the visual perception of people. The cloud top height data describe the horizontal and vertical distribution of the cloud, which ensures the accuracy of the cloud distribution. At the same time, the optical thickness approximately reflects the cloud cover. The combination of the two can reflect the objective distribution of cloud particles.

In this paper, a particle generation method with selection criteria is adopted: load the cloud top height data and convert it into a one-dimensional array A. The whole three-dimensional space of cloud distribution is divided into $M \times N$ grids horizontally. Generate random number R as the index in A, and calculate the grid where the particles are based on R

$$\text{col}=\text{MOD}(R,\text{COLS}). \tag{3}$$

$$\text{row}=\text{floor}(R/\text{COLS}). \tag{4}$$

where col and row respectively represents the column and row in the grid. The column and row number of the grid is multiplied by the side length of the grid to get the plane position of the particle.

In the process of generating random numbers, the following methods are used to make the optical thickness interfere with the particle distribution: the random number is generated and used as the index in A to obtain the corresponding cloud optical thickness data after the conversion of the resolution. According to the following formula

$$M = \lambda_i \times \frac{K}{\lambda_{\max}} - \tau, \tag{5}$$

where λ represents the current cloud optical thickness value, $\lambda_{\max}$ represents the maximum cloud optical thickness, and K is a constant, which is used to control the probability of a random number being selected. τ is a random number from 0 to 1. When M is more than 0, the random number is retained, and when M is less than 0, the random number is generated again until M is more than 0. It can be seen from (5) that where the cloud optical thickness is large, the probability of randomly generated particles being selected is high, and the smaller the cloud optical thickness is, the smaller the probability of being selected is.

2.3 Simulation of cloud evolution

In the real world, clouds will drift and spread, due to the influence of natural factors such as high airflow. Therefore, simulating the dynamic behavior of clouds is the key to enhancing the realism of cloud scenes. Cloud motion is affected by four factors: pressure, diffusion, advection, and external forces. Affected by the above factors, cloud particles generate acceleration and thus form dynamic behaviors such as fluttering.

The WRF model data itself contains the evolution of weather over a time series, with an initial velocity and acceleration when the particles are initialized. As the timeline moves forward, the positions of the particles change, allowing the cloud to change dynamically. According to the laws of Newtonian mechanics, in the process of rendering each frame in this paper, the new position of particles is calculated by the following formula

$$S = S_0 + V_0 T + \frac{1}{2} a T^2. \tag{6}$$

Formula (6) is the calculation method of any coordinate component of 3D coordinates system, where S is the coordinate component, S_0 is the initial value of the coordinate component, V_0 is the velocity component, a is the acceleration component, and T is the time in the current life cycle.

2.4 Display with the distance

In the process of displaying complex models, the graphics that need to be highlighted will occupy more pixels on the screen, while those that do not need to be highlighted will not be accurately drawn with a large number of pixels. LOD technology is based on a series of evaluation criteria to select different precision to achieve the object of multi-resolution display [26]. Generally, the LOD technology in the traditional geometric modeling means that the number of vertices in a geometric model is determined by the distance from the object to the viewpoint [27–29]. For the particle system, when the viewpoint is close, more particles are generated; on the contrary, fewer particles. Thus, less running time is occupied and the rendering efficiency is improved. How to formulate the ratio of the number of particles with the distance is the focus of this section. Inspired by the principles of camera imaging, we derived the proportion value between the size of the object on the imaging surface and the actual size, which is taken as the proportion value of the number of vertices in the simplified geometric model to the number of vertices in the original model. The law states that

$$\eta = 1 - \frac{f}{L}. \tag{7}$$

In the three-dimensional coordinate system, the direction is considered, (7) is turned into (8)

$$\eta = 1 - \left(\frac{f}{L}\right)^3. \tag{8}$$

According to the characteristics of particle system, the following method can be derived. Then,

$$f = L_0. \tag{9}$$

When the distance reaches L_0, the simplification rate reaches the lower limit, and a further reduction will not be carried out. As for the upper limit of simplification rate, it can be 1 theoretically. However, it is often simplified to stop when only one particle is produced. Since modeling simplification is often hierarchical, the simplification rate η of the m-level model can be expressed as

$$\eta_m = 1 - \frac{N_m}{N_0}, \tag{10}$$

where N_0 is the particle number of the initial model and N_m is the number of m-level model. The number of particles generated in the distance L_m can be used as N_m, and the mean number of particles in the distance L_0 can be used as N_0. The relationship between N_m and L_m can be deduced as follows

$$\eta_m = 1 - \frac{N_m}{N_0} = 1 - \frac{L_0}{L_m}. \tag{11}$$

$$N_m = \frac{N_0 \times L_0}{L_m}. \tag{12}$$

Similar to (8)

$$N_m = N_0 \times \left(\frac{L_0}{L_m}\right)^3. \tag{13}$$

2.5 Illumination model

When light passes through the medium, the energy of light will be attenuated to different degrees due to the absorption and scattering of light by the medium [30]. Clouds observed in the physical world are made up of a huge number of condensed water droplets, which absorb and scatter light as it passes through them. The scattered light will be absorbed and scattered as it passes through other particles. Therefore, the main work of this section is how to simulate the attenuation of light energy by cloud particles, so that the light intensity and direction on different cloud particles are different, and the final expression is the brightness of the particles. The propagation of light is attenuated by increasing distance. The brightness of light is inversely proportional to the distance from the light source. It is shown that

$$i = \frac{I}{d^2}, \tag{14}$$

where i is the light intensity at the sampling point, d is the distance, and I is the light intensity. What is obtained here is the brightness value at a sampling point. In order to obtain the brightness value of all the sampling points, it is necessary to integrate and sum the brightness values of all the sampling points, which is described in Fig. 2.

S is the position of the light source, C is the position of the camera (viewpoint), O is the position of the object, t_1, t_2, t_3, and t_4 are the four sampling points, where function $L(t)$ is the brightness value at the sampling point t. Function $x(t)$ is the integral curve of the line of sight. The formula can be obtained as follows:

$$x(t) = r_c + t \cdot r_d, \tag{15}$$

where r_c is the line of sight, r_d is the direction of sight, and the function $x(t)$ is substituted into $L(t)$ to get

$$L(t) = \frac{I}{[x(t) - s]^2}, \tag{16}$$

where s is the position of the light source, and the cloud depth is d. We need to obtain the integral from 0 to d of $L(t)$, which can be obtained from the above two equations

$$L(t) = \frac{I}{v \cdot \left[\arctan\left(\frac{b+d}{v}\right) - \arctan\left(\frac{b}{v}\right)\right]^2}, \tag{17}$$

where

$$v = \frac{1}{\sqrt{c - b^2}}, b = r_d \cdot (r_c - s), c = (r_c - s)^2. \tag{18}$$

The formula for calculating the illumination intensity of each particle is obtained. The particles at the boundary of the cloud are illuminated directly by the light source without any shielding, so these particles should be brighter. Points within the cloud layer are different in depth from

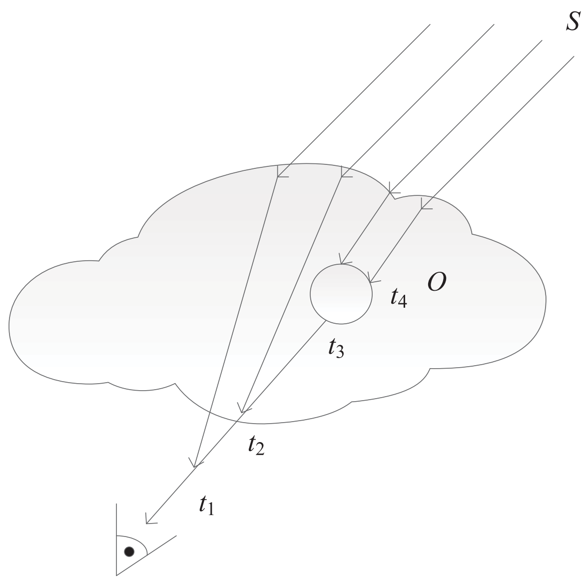

Fig. 2 Sum of sample point brightness values

the boundary of the cloud cluster, and the light intensity calculated according to formula (17) is also different. According to the light intensity, a corresponding gray value is assigned.

3 Experiment and evaluation

3.1 Experiment settings

The implementation is done on a PC with an NVIDIA GeForce GTX 1080Ti Founders Edition. Its basic configuration consists of screen resolution 1920 × 1080, RAM 8.00 GB, and Windows 7 operating system. C++ was used to complete the rendering process, and OpenGL was used to assist the visualization.

3.2 The experimental process

Four steps make up our experimental process:

(1) Traversing the data of top cloud, the data points with empty filling values are removed, where the area grid without cloud. Get a one-dimensional array of valid values.

(2) Read the data of cloud optical thickness, generate cloud particles according to the method in Section 2.2, initialize the initial position of each particle, and determine the distribution of particles according to the cloud optical thickness.

(3) The wind field model is introduced to calculate the airflow direction and velocity at the position of particles, further calculate the acceleration of each particle, and assign an initial velocity and life cycle length to the particle. When the wind field model is introduced, it is introduced in the horizontal direction and the vertical direction respectively.

(4) Parallel light source is added to simulate sunlight. When rendering frame by frame, their positions are not fixed due to the dynamic characteristics of cloud particles, so it is necessary to recalculate the light intensity at the positions of all particles. Suppose that the current particle is particle A, and particle B is the particle closest to the light source between particle A and the light source (within a certain range), then the distance between particle A and B in the direction of the light source is taken as the depth of particle A from the cloud boundary. According to formula (18), the light intensity at the depth is calculated as well as the color and transparency of the particles.

3.3 Results and evaluation

In this section, we focus on analyzing the performance of our proposed method and benchmark under the same configuration. The results of the experiments are shown here. The process of simulating the cloud with particles is shown in Fig. 3a–f, where the number, radius, and distribution of spherical particles are determined according to the optical thickness of the cloud. Meanwhile, particles'

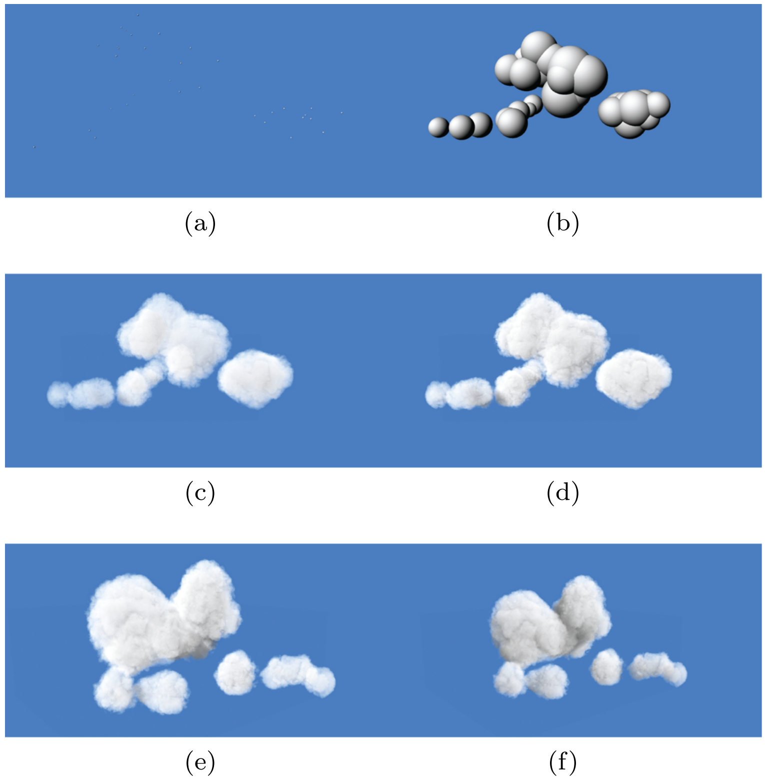

Fig. 3 Step for partical system to simulate clouds. **a** Particles of one cloud. **b** Basic shape. **c** Add texture of cloud. **d** Add more details. **e** Simplification. **f** Illumination

quantity, FPS, and CPU utilization are analyzed to verify its validity, compared with benchmark.

3.3.1 Steps for particle system to simulate clouds

Figure 3 unfolds the process of simulating clouds with our proposed method. It can be seen in Fig. 3a that many spherical particles form the basic shape of a cloud. Then, the WRF data is used to obtain the color, shape, and other properties of the particles, so that the particles show the characteristics of fuzzy surface of cloud. According to the method in Section 2.4 above, the simplification rate η is introduced. In Fig. 3c, η is 0.84 while it is 0.17 in Fig. 3d. Distinctly, it can be found that cloud in Fig. 3d has more details and distinct texture features than cloud in Fig. 3c. In Fig. 3e, cloud illumination is introduced but not in Fig. 3f, according to Section 2.5. It can be estimated from the two figures that Fig. 3e has a stronger three-dimensional sense and is more realistic after the introduction of light. The above results indicate that the method in this paper can realize the simulation of clouds based on WRF data. In Fig. 4, four clouds using our simulation method are given, and it is clear that they show every feature of the real-word clouds.

3.3.2 Comparison of particles' quantity

In this section, N_0 is set as 5×10^3, L_0 is set as 10. Next, and L_m regularly ranges from 25 to 65. The interval of L_m range is 15. As can be seen from the Fig. 5, after using the proposed simplification method, the number of particles produced each time gradually decreases while the distance becomes longer. When it goes to 50, it does not simplify anymore. Hence, 65 corresponds to the same number of particles as the previous one.

3.3.3 Comparison of FPS

In this subsection, the same clouds were simulated from different distance viewpoints. And FPS was recorded at different distances. As can be seen in Fig. 6, under the same circumstance, FPS decreases gradually with the increase of particle number either before or after simplification, which proves that lots of particles generated to use a lot of computing resource. Technologically, it can be seen that FPS is always higher after the simplification.

3.3.4 Comparison of CPU utilization

By comparing the experimental results, it can be found that the three-dimensional cloud image simulated by the algorithm in this paper has a better effect in detail and is

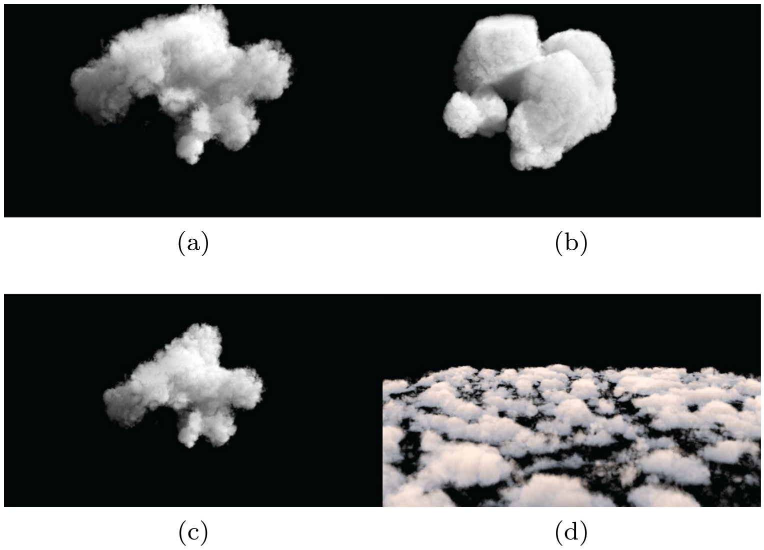

Fig. 4 Four clouds drawn using the proposed method

more accurate. In order to verify the superiority of this algorithm in CPU utilization, different numbers of particles are generated. In this way, the above Fig. 7 shows the CPU utilization is lower after the simplification.

From the above analysis, the cloud generated by the cloud data visualization system has a good visualization effect, and the application of LOD technology in the visualization system can effectively improve the real-time performance of the visualization system, which plays an important role in the performance of the system.

4 Related work

To simulate the 3D clouds, it is crucial that the simulation results have the laws and characteristics of real clouds [31]. In order to obtain realistic effects, researchers have proposed a variety of modeling methods, which are generally divided into two categories: methods based on individual growth and methods based on physical processes. Typical methods based on individual growth are mainly managed to obtain the visual characteristics and enhance the realism of clouds without taking the physical process into account, such as particle system, metaball, fractal method, texture method, and noise- and cloud-driven method [32]. Methods based on physical processes—the numerical simulation method—are mainly to solve the fluid dynamics equation—Navier-Stokes. The most representative method was the Harris, which solved the numerical equation including the cloud motion equation,

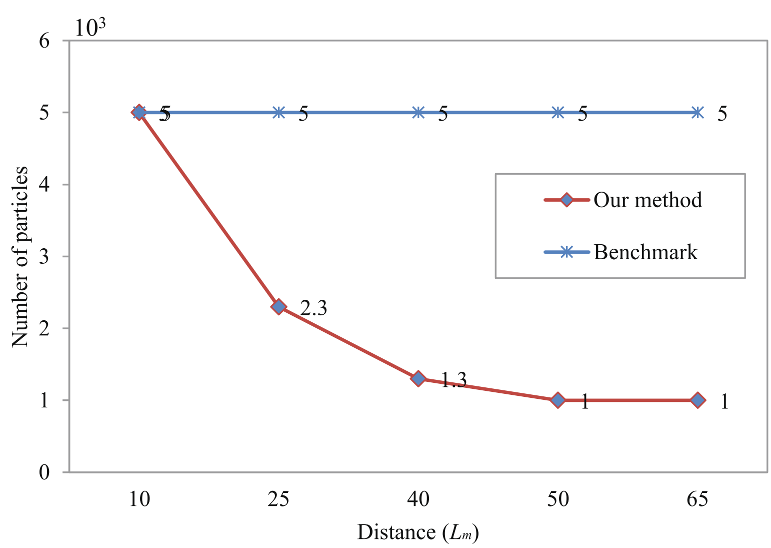

Fig. 5 Comparison of particles' quantity using benchmark and our method

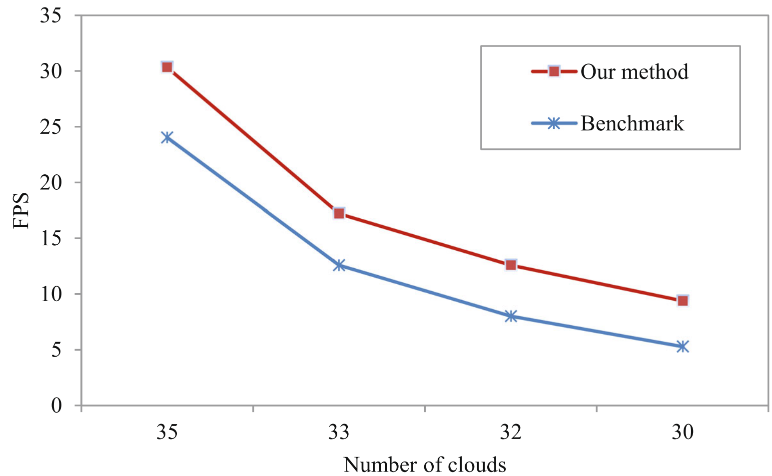

Fig. 6 Comparison of FPS using benchmark and our method

the water balance equation, and the thermodynamic equation [33].

These methods could vividly simulate the dynamics of the cloud, and the effect is realistic, but the computation is large [34]. Moukalled [35] described the dynamic changes of cloud through a set of Boolean rules in cellular automata method and used Sigmoid function to convert the discrete results into continuous density distribution, thus improved the real-time performance of cloud simulation. These methods do not need to simulate the real physical process of cloud growth and are fast in drawing and rendering, so they are suitable for the simulation of small and medium-sized virtual cloud. Prashant Goswami adopted a physical method to intuitively control the entire life cycle of the cloud through the calculation of parameters such as pressure and viscosity between the clouds [36, 37]. Lots of computing has reduced the rendering details of the cloud as well as the visual effect.

According to the classification of clouds, the ambient light, diffuse light, and specular reflection light were introduced to process the light and shadow of clouds by Luo [38] and realized the cloud simulation at different times of the day. Meanwhile, it constructed and simulated ten kinds of representative clouds. The method has achieved good simulation effects; however, the sense of reality and real-time of cloud simulation need to be improved. A different strategy was employed by Chulichkov et al. [39–41] to divide clouds according to the different morphological characteristics of clouds at different heights. The strategy is to map the detailed characteristics of different types of clouds by using different simulation methods in different cloud regions under the condition

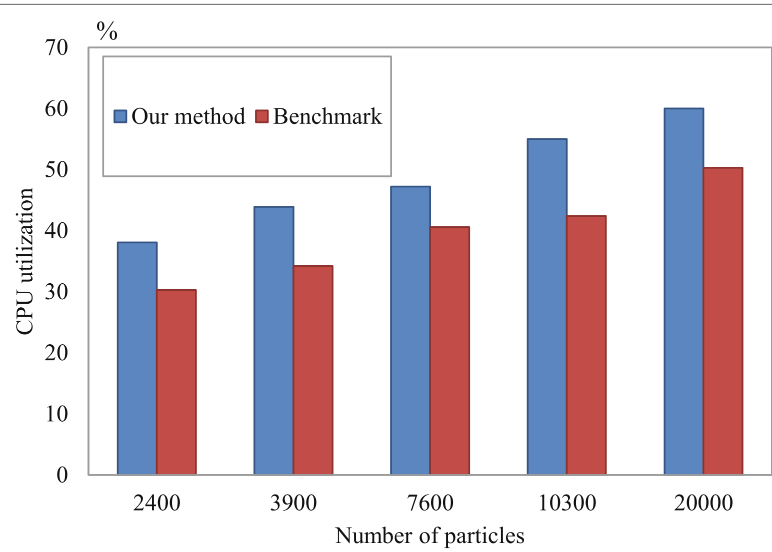

Fig. 7 Comparison of CPU utilization using benchmark and our method

of altitude. In addition, on the premise of not changing the overall movement trend of the cloud, a velocity function related to cloud position and viewpoint distance is constructed and added to the original velocity field to enhance the floating effect of the cloud, so as to achieve a more realistic simulation effect. There are many factors influencing the morphological characteristics of clouds. This method only considered the altitude as the simulation condition, which not only simplifies the simulation process, but also inevitably sacrifices the detailed characteristics of clouds.

The limitations of above methods are that these methods simply make simplifying assumptions about real environmental phenomena based on physical equation and other parameters of numerical simulation and the input data is not the real meteorological data and thus does not belong to the true sense of real 3D simulation of cloud. Moreover, few feasible methods are raised for generating clouds in interactive frame rates from the weather forecast data.

5 Conclusion and future work

In this paper, we have studied simulation methods and graphic algorism of irregular fuzzy objects, such as clouds, flowing water, and snow. Specifically, a particle system of three-dimensional cloud based on WRF data has been established. Then, a simplification rate has been calculated to simplify the particle system and improve the speed of modeling and rendering. At last, experimental evaluations have been carried out to verify the validity of our proposed method.

There are quantities of ways in which various meteorological factors are related to each other, and it is tough to establish a model according to small accounts of data. Moreover, clouds generated by data visualization system has good visual effect, but compared with the actual clouds, there is a large gap, and many clouds' details are ignored or changed. In the future, more factors which can determine simulation results of cloud data will also be taken into consideration.

Abbreviations

LOD: Level of detail; WRF: Weather research and forecasting

Acknowledgements

This research is supported by the National Natural Science Foundation of China under grant no. 41675155.

Authors' contributions

YX, XK, and PL conceived and designed the study. YX and XK performed the simulations. XK wrote the paper. All authors reviewed and edited the manuscript. All authors read and approved the final manuscript.

Availability of data and materials

The dataset supporting the conclusions of this article is available, which can downloaded at https://drive.google.com/open?id=1pThqWafKfniPwqCcFqe-JoIZPIVrEW2r.

Competing interests

The authors declare that they have no competing interests.

Received: 1 July 2019 Accepted: 23 October 2019
Published online: 20 November 2019

References

1. W. T. Reeves, *Particle systems—a technique for modeling a class of fuzzy objects*, (1998). https://doi.org/10.1145/800059.801167
2. M. Mohan, A. P. Sati, WRF model performance analysis for a suite of simulation design. At. Res. **169**, 280–291 (2016)
3. J. Zhang, N. Xie, X. Zhang, K. Yue, W. Li, D. Kumar, Machine learning based resource allocation of cloud computing in auction. Comput. Mater. Continua. **56**(1), 123–135 (2018)
4. X. Xu, Y. Chen, X. Zhang, Q. Liu, X. Liu, L. Qi, A blockchain-based computation offloading method for edge computing in 5G networks. Softw. Pract. Experience. https://doi.org/10.1002/spe.2749
5. M. Chakroun, S. Bastin, M. Chiriaco, H. Chepfer, Characterization of vertical cloud variability over Europe using spatial lidar observations and regional simulation. Clim. Dyn., 1–23 (2016). https://doi.org/10.1007/s00382-016-3037-3
6. A. P. Biazar, A. White, R. McNider, M. Khan, B. Dornblaser, Y. Wu, in *EGU General Assembly Conference Abstracts, vol. 19*. Satellite cloud assimilation in the weather research & forecasting (wrf) model and its impact on air quality simulations, (2017), p. 10641
7. X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, W. Dou, Become: blockchain-enabled computation offloading for IOT in mobile edge computing. IEEE Trans. Ind. Inform. (2019). https://doi.org/10.1109/tii.2019.2936869
8. P. Goswami, F. Neyret, in *Proceedings of the 19th Symposium on Interactive 3D Graphics and Games*. Real-time landscape-size convective clouds simulation (ACM, 2015), pp. 135–135. https://doi.org/10.1145/2699276.2721396
9. L. Qi, R. Wang, C. Hu, S. Li, Q. He, X. Xu, Time-aware distributed service recommendation with privacy-preservation. Inform. Sci. **480**, 354–364 (2019)
10. W. Li, X. Liu, J. Liu, P. Chen, S. Wan, X. Cui, On improving the accuracy with auto-encoder on conjunctivitis. Appl. Soft Comput., 105489 (2019). https://doi.org/10.1016/j.asoc.2019.105489
11. Y. Kogan, A cumulus cloud microphysics parameterization for cloud-resolving models. J. Atmos. Sci. **70**(5), 1423–1436 (2013)
12. X. Xu, X. Zhang, M. Khan, W. Dou, S. Xue, S. Yu, A balanced virtual machine scheduling method for energy-performance trade-offs in cyber-physical cloud systems. Futur. Gener. Comput. Syst. (2017). https://doi.org/10.1016/j.future.2017.08.057
13. Y. Yuan, Y.-S. Ong, A. Gupta, H. Xu, Objective reduction in many-objective optimization: evolutionary multiobjective approaches and comprehensive analysis. IEEE Trans. Evol. Comput. **22**(2), 189–210 (2017)
14. L. Qi, Y. Chen, Y. Yuan, S. Fu, X. Zhang, X. Xu, *A QoS-aware virtual machine scheduling method for energy conservation in cloud-based cyber-physical systems*, (2019), pp. 1–23. https://doi.org/10.1007/s11280-019-00684-y
15. S. Wan, Y. Zhao, T. Wang, Z. Gu, Q. H. Abbasi, K. Kwang, R. Choo, Multi-dimensional data indexing and range query processing via Voronoi diagram for internet of things. Fut. Gener. Comput. Syst. **91**, 382–391 (2019)
16. L. Wang, H. Zhen, X. Fang, S. Wan, W. Ding, Y. Guo, A unified two-parallel-branch deep neural network for joint gland contour and segmentation learning. Fut. Gener. Comput. Syst. **100**, 316–324 (2019)
17. W. W. Grabowski, H. Morrison, S.-i. Shima, G. C. Abade, P. Dziekan, H. Pawlowska, Modeling of cloud microphysics: can we do better? Bull. Am. Meteorol. Soc. (2018). https://doi.org/10.1175/bams-d-18-0005.1
18. J. Liu, W. Wang, D. Li, S. Wan, H. Liu, Role of gifts in decision making: an endowment effect incentive mechanism for offloading in the IoV. IEEE Internet Things J. (2019). https://doi.org/10.1109/jiot.2019.2913000
19. F. Biljecki, H. Ledoux, J. Stoter, An improved LOD specification for 3D building models. Comput. Environ. Urban Syst. **59**, 25–37 (2016)
20. X. Xu, Q. Liu, Y. Luo, K. Peng, X. Zhang, S. Meng, L. Qi, A computation offloading method over big data for IoT-enabled cloud-edge computing. Futur. Gener. Comput. Syst. **95**, 522–533 (2019)
21. Q. Y. Pan, B. I. Shuo-Ben, L. U. Liang-Hu, Y. Shi, Fast algorithm based on particle system for simulating 3D dynamic clouds. J. Syst. Simul. (2014)

22. M. C. G. Ooi, A. Chan, S. Kumarenthiran, K. I. Morris, M. Y. Oozeer, M. A. Islam, S. A. Salleh, *Comparison of WRF local and nonlocal boundary layer Physics in Greater Kuala Lumpur, malaysia*, (2018), p. 012015. https://doi.org/10.1088/1755-1315/117/1/012015
23. H. Liu, L. Wang, Qun. D.U., An overview of recent studies on atmospheric boundary layer physics (2012–2017). Chin. J. Atmos. Sci. (2018)
24. X. Xu, Y. Li, T. Huang, Y. Xue, K. Peng, L. Qi, W. Dou, An energy-aware computation offloading method for smart edge computing in wireless metropolitan area networks. J. Netw. Comput. Appl. **133**, 75–85 (2019)
25. R. J. A. Putri, T. Setyawan, Identification of atmospheric boundary layer thickness using doppler radar datas and WRF - ARW model in merauke. IOP Conf. Ser. Earth Environ. Sci. **54** (2017). https://doi.org/10.1088/1755-1315/54/1/012101
26. S. Donkers, H. Ledoux, J. Zhao, J. Stoter, Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings. Trans. Gis. **20**(4), 547–569 (2016)
27. Z. Gao, H.-Z. Xuan, H. Zhang, S. Wan, K.-K. R. Choo, Adaptive fusion and category-level dictionary learning model for multi-view human action recognition. IEEE Int. Things J. (2019). https://doi.org/10.1109/jiot.2019.2911669
28. S. Ding, S. Qu, Y. Xi, S. Wan, A long video caption generation algorithm for big video data retrieval. Futur. Gener. Comput. Syst. **93**, 583–595 (2019)
29. Y. Yuan, H. Xu, B. Wang, B. Zhang, X. Yao, Balancing convergence and diversity in decomposition-based many-objective optimizers. IEEE Trans. Evol. Comput. **20**(2), 180–198 (2015)
30. E. Yusov. High-performance rendering of realistic cumulus clouds using pre-computed lighting, (2014), pp. 127–136
31. X. Xu, Y. Xue, L. Qi, Y. Yuan, X. Zhang, T. Umer, S. Wan, An edge computing-enabled computation offloading method with privacy preservation for internet of connected vehicles. Futur. Gener. Comput. Syst. **96**, 89–100 (2019)
32. M. Gavaises, F. Villa, P. Koukouvinis, M. Marengo, J. P. Franc, Visualisation and les simulation of cavitation cloud formation and collapse in an axisymmetric geometry. Int. J. Multiphase Flow. **68**(s 3–4), 14–26 (2015)
33. A. A. Jensen, J. Y. Harrington, Modeling ice crystal aspect ratio evolution during riming: a single-particle growth model. J. Atmos. Sci. **72**(7), 150407121742006 (2015)
34. X. Xu, S. Fu, L. Qi, X. Zhang, Q. Liu, Q. He, S. Li, An IoT-oriented data placement method with privacy preservation in cloud environment. J. Netw. Comput. Appl. **124**, 148–157 (2018)
35. F. Moukalled, L. Mangani, M. Darwish, The finite volume method in computational fluid dynamics. Fluid Mech. Appl. **29**(1), 3–38 (2016)
36. P. Goswami, F. Neyret, Real-time landscape-size convective clouds simulation. Acm Interact. Graph., 135–135 (2016). https://doi.org/10.1145/2699276.2721396
37. Y. Yuan, W. Banzhaf, ARJA: automated repair of java programs via multi-objective genetic programming. arXiv preprint (2017). arXiv:1712.07804
38. R. Luo, W. Liao, H. Zhang, L. Zhang, P. Scheunders, Y. Pi, W. Philips, Fusion of hyperspectral and LiDaR data for classification of cloud-shadow mixed remote sensed scene. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **PP**(99), 1–14 (2017)
39. A. I. Chulichkov, M. S. Andreev, G. S. Golitsyn, N. F. Elansky, A. P. Medvedev, O. V. Postylyakov, On cloud bottom boundary determination by digital stereo photography from the earth's surface. Atmos. Ocean. Opt. **30**(2), 184–190 (2017)
40. Z. Gao, D. Y. Wang, S. H. Wan, H. Zhang, Y. L. Wang, Cognitive-inspired class-statistic matching with triple-constrain for camera free 3D object retrieval. Futur. Gener. Comput. Syst. **94**, 641–653 (2019)
41. C. W. F. Barbosa, Y. Dobashi, T. Yamamoto, Adaptive cloud simulation using position based fluids. Comput. Animat. Virtual Worlds. **26**(3-4), 367–375 (2015)

Publisher's Note