# An Optimization Method for Computer Animation Develpoment Projects

Yan Ren[1,*]

1 *The Media Department*
Hebei Vocational College of Art
Shijiazhuang, Hebei, China

***Abstract* — Because the optimization of computer animation development projects is an important factor which cannot be ignored, different optimization methods have different costs: the use of good quality computing resources is high, while the use of common resources is low. Users make their choices based on consideration of budget as well as waiting time. To solve the problem, we propose a double fitness particle swarm optimization algorithm (DFPSO) based on time and cost constraints. In the algorithm, both the complete time of all tasks and the cost in computer animation trajectory planning control are used as scheduling objectives. Through this algorithm, the proposed optimization method not only shortens total task completion time but also costs less.**

***Keywords - optimization method; computer animation trajectory planning control; computing resources***

## I. INTRODUCTION

It is an important issue for the trajectory planning control to how to reasonably allocate the computing resources and balance resources' load. Existing optimization for computer animation trajectory algorithms for trajectory planning control are not well to take into account the load balancing problem for the pursuit of the shortest completion time. It may lead to the phenomenon of unbalanced resources' load. To solve this problem, a double-fitness particle swarm optimization (DFPSO) based on resource pre-classification is proposed in this paper [1-2].

In the new algorithm, the resources classifies through the information of measuring the ability of computing and communications, and then calculate the product of the expected execution time of tasks in the resource and its corresponding resource class, which is regarded as another fitness function of optimization for computer animation trajectory [3]. The results generated by this algorithm not only make the task completion time shorter, but also have a higher utilization of system resources, which are taking into account the minimum execution time and load balancing. The simulation shows that DFPSO is an efficient optimization for computer animation trajectory algorithm in the trajectory planning control by contrast with the conventional particle swarm optimization (PSO) [4-5].

Optimization for computer animation trajectory in trajectory planning data center is the core issue of trajectory planning control, also the key technology for trajectory planning control's large-scale application and system performance improve. Advanced optimization for computer animation trajectory has a great significance for cloud service providers to improve the efficiency of computing resources, save energy, improve resource sharing, and reduce operating costs deserves the further systematic study.

For the Internet, trajectory planning control appears to be a revolution. It is widely used because of its characteristics of large-scale, virtualization, reliability, cheapness, and so on. Trajectory planning control can provide three services of IaaS, PaaS and SaaS of which, IaaS is the basis of the other two services.We study IaaS to provide better service for the others. IaaS service can be provided by building open-source platforms, among them C1oudStack is better in many aspects such as usage, easy-deploying, easy-studying, scheduling policy and so on, so we choose this platform as the research object [6-9].

Based on the analysis of the C1oud Stack internal resource scheduling mechanism, we find that the two layers of the resources scheduling plays an important role in the task optimal span and load balancing. For the first layer, the virtual machine to the physical machine deployment strategy determines the resource utilization and load imbalance; while for the second layer, the task to the virtual machine allocation strategy determines the optimal span of the task execution time. In this paper, we use the Particle Swarm Optimization (PSO) to optimize the resource scheduling of the two layers. Because the particle swarm algorithm has the advantage of high precision and fast convergence speed, it can fulfill the requirements of the scheduling and shorten operation time, so it is the ideal scheduling algorithm in trajectory planning control environment. To deal with the problem of PSO's premature, Simulated Annealing (SA) is used to optimize PSO.

To deal with the two layers in resource scheduling, we present a virtual machine deployment algorithm based on improved PSO and dual-fitness optimization for computer animation trajectory algorithm based on PSO, and then the C1oudSim simulation tool is used to carry out the simulation, with results showing that the proposed algorithm can effectively improve the optimal span and optimization of load balancing. In order to make the studied algorithm have practical applications, through the study of cloud stack's source code, we find where the code structure of virtual machine deployment and optimization for computer animation trajectory locates in this open source cloud platform framework. Aiming at this problem, we present a method to improve the internal resource scheduling strategy in C1oudStack. For the internal optimization for computer animation trajectory in C1oudStack, a new optimization for computer animation trajectory tool is used to upgrade the original way of optimization for computer animation trajectory, and to increase new functions under the premise of not changing original optimization for computer animation trajectory function in cloudStack.

## II. THEORETICAL FRAMEWORK

As the next generation application model of the Internet, trajectory planning control is also a business model. Faced with a large number of users, trajectory planning control need to process thousands of data and tasks in a short time, the level of trajectory planning control platform plays a decisive role in it. To a large extent, optimization for computer animation trajectory algorithm affects the performance of the trajectory planning control platform, so improving the optimization for computer animation trajectory algorithm becomes a research hotspot. The research of the optimization for computer animation trajectory and resource allocation is few, and the existing optimization for computer animation trajectory algorithms for trajectory planning control usually laid their attention on the pursuit of the shortest completion time, however, they are not well to take into account the cost of all the tasks for the pursuit of the shortest completion time. The cost of all the tasks is an important factor which cannot be ignored. Different computing resources in cloud platform have different costs; the use of excellent computing resources is higher, while the use of common resources is lower. The cloud users usually make their choices based on comprehensive consideration of the economic budget as well as the wait time. To solve the problem, a double fitness particle swarm optimization algorithm (DFPSO) based on the time and cost constraints are proposed in this paper. In the algorithm, both the complete time of all tasks and the cost of all tasks are used as scheduling objectives. Through this algorithm, the proposed optimization for computer animation trajectory not only shortens total task completion time and also costs less.

As a new business model, trajectory planning control has the characteristics of grid computing, distributed computing, and so on.

(1) Virtualization. Trajectory planning control virtualization technology is used to integrate software and hardware resources into services, users can use the network to use and access to these services, without knowing its specific operation mode and location, just as a PC or a PDA such as trajectory planning control terminal equipment, can easily get the required services, supercomputing such services are also included in the [7].

(2) Dynamic scalability. The scale of the cloud can be dynamically scalable, able to provide functionality quickly and flexibility to achieve expansion, in order to meet the needs of the application and user scale growth.

(3) General purpose. Trajectory planning control is not specific to a particular application, it has no relationship with the physical platform, in the same cloud can be constructed from a wide variety of applications, and can support different applications running.

(4) High reliability. Due to the cloud using the data redundancy, data replica fault-tolerant, VM dynamic migration technology as well as computing nodes isomorphic interchangeable is measured to ensure that the service of high reliability, so it is more feasible than the local reliable calculation, user service is not affected because of a machine failure.

(5) On demand. The user can buy the resources in the cloud according to their own needs.

(6) On-demand deployment. With virtualization technology, the deployment of cloud platform has become very flexible, it can change the capacity according to the needs of the user, and the application is deployed to the corresponding virtual resource pool to provide users with services.

(7) High cost performance. Because of the special fault tolerance measures, the server and node in the trajectory planning control are relatively low, and the centralized management makes the management cost of the trajectory planning data centre is significantly lower than before. Users only need to spend less money to complete the task than before.

Now the development of trajectory planning control is not a short duration of time out, but after decades of accumulation and development of the computer technology. It is the fusion of many kinds of traditional calculation and distillation, as shown in figure 1. In the following, the paper briefly introduces the differences between the calculation of the trajectory planning control, the grid computing, the utility computing, and the parallel computing.

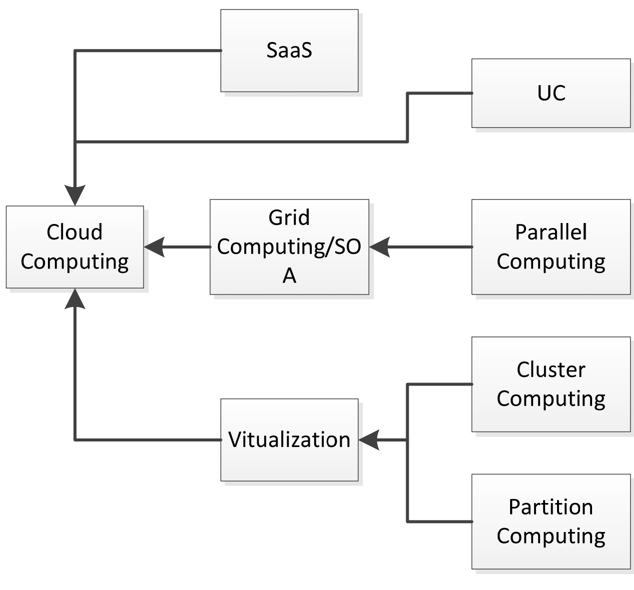



Figure 1. The relationship between trajectory planning control and traditional computing.

## III. THE ALGORITHM

The algorithm based on fractal theory can be expressed as following:

$$f^{(\alpha)}(x0) = \left.\frac{df(x)}{dx^{\alpha}}\right|_{x=x_0} = \lim_{\delta x\to 0}\frac{\Delta^{\alpha}(f(x)-f(x_0))}{(x-x_0)^{\alpha}} \quad (1)$$

for $0 < a \le 1$ where

$$\Delta^{\alpha}(f(x)-f(x_0)) \cong \Gamma(1+\alpha)\lim_{x\to\infty}\Delta(f(x)-f(x_0)) \quad (2)$$

And local fractional integral of $f(x)$ defined by Eq.3.

$$\begin{aligned} {}_aI_b^{(\alpha)} f(t) &= \frac{1}{\Gamma(1+\alpha)}\int_a^b f(t)(dt)^{\alpha} \\ &= \frac{1}{\Gamma(1+\alpha)}\lim_{\Delta t\to 0}\sum_{j=0}^{j=N-1} f(t_j)(\Delta t_j)^{\alpha} \end{aligned} \quad (3)$$

With $\Delta t_j = t_{j+1} - t_j$ and $\Delta t = \max\{\Delta t_1, \Delta t_2, \cdots, \Delta t_j, \cdots\}$, where for $j = 1, 2, \cdots, N-1$, $[t_j, t_{j+1}]$ and $t_0 = a$, $t_N = b$.

If $f(x)$ is defined on the real line $-\infty < x < \infty$, its local fractional Hilbert transform, denoted by $f_x^{H,\alpha}(x)$ is defined by

$$H_{\alpha}\{f(t)\} = \hat{f}_H^{\alpha}(x) = \frac{1}{\Gamma(1+\alpha)}\int_R \frac{f(t)}{(t-x)^{\alpha}}(dt)^{\alpha} \quad (4)$$

Where $x$ is real and the integral is treated as a Canchy principal value, that is,

$$\begin{aligned} &\frac{1}{\Gamma(1+\alpha)}\int_R \frac{f(t)}{(t-x)^{\alpha}}(dt)^{\alpha} \\ &= \lim_{\varepsilon\to 0}\Big[\frac{1}{\Gamma(1+\alpha)}\int_{-\infty}^{x-\varepsilon}\frac{f(t)}{(t-x)^{\alpha}}(dt)^{\alpha} + \\ &\frac{1}{\Gamma(1+\alpha)}\int_{x+\varepsilon}^{\infty}\frac{f(t)}{(t-x)^{\alpha}}(dt)^{\alpha}\Big] \end{aligned} \quad (5)$$

To obtain the inverse local fractional Hilbert transform, write again Eq. (4) as

$$\begin{aligned} \hat{f}_H^{\alpha}(x) &= \frac{1}{\Gamma(1+\alpha)}\int_{-\infty}^{\infty}\frac{f(t)}{(t-x)^{\alpha}}(dt)^{\alpha} \\ &= \frac{1}{\Gamma(1+\alpha)}\int_{-\infty}^{\infty} f(t)g(x-t)(dt)^{\alpha} = f(x)*g(x), \end{aligned} \quad (6)$$

The equation of motion is as follows:

$$\partial_j(C_{ijkl}\partial_k u_l + e_{kij}\partial_k\varphi) - \rho\ddot{u}_i = 0 \quad (7)$$

Under the linear theory, that is:

$$\partial_j(e_{ijkl}\partial_k u_l - \eta_{kij}\partial_k\varphi) = 0 \quad (8)$$

In piezoelectric media, linear equation can be expressed into the following simplified forms:

$$L(\nabla,\omega)f(x,\omega) = 0,\ \ L(\nabla,\omega) = T(\nabla) + \omega^2\rho\mathrm{J} \quad (9)$$

In which,

$$T(\nabla) = \begin{Vmatrix} T_{ik}(\nabla) & t_i(\nabla) \\ t_k^T(\nabla) & -\tau(\nabla) \end{Vmatrix},\ \mathrm{J} = \begin{Vmatrix} \delta_{ik} & 0 \\ 0 & 0 \end{Vmatrix},$$

$$f(x,\omega) = \begin{Vmatrix} u_k(x,\omega) \\ \varphi(x,\omega) \end{Vmatrix} \quad (10)$$

Consider delay, the L can be expressed as:

$$L^0 = \begin{Vmatrix} C_{ijkl}^0 & e_{kij}^0 \\ e_{ikl}^{0T} & -\eta_{ik}^0 \end{Vmatrix} \quad (11)$$

These functions can be expressed in the following form:

$$C(\mathrm{x}) = C^0 + C^1(\mathrm{x}),\ \ e(\mathrm{x}) = e^0 + e^1(\mathrm{x}),$$

$$\eta(\mathrm{x}) = \eta^0 + \eta^1(\mathrm{x}),\ \ \rho(\mathrm{x}) = \rho_0 + \rho_1(\mathrm{x}) \quad (12)$$

The value with superscript of 1 represents the difference below:

$$C^1 = C - C^0,\ e^1 = e - e^0,$$

$$\eta^1 = \eta - \eta^0,\ \rho_1 = \rho - \rho_0 \quad (13)$$

The whole function can be simplified into the following integral equation set:

$$\begin{aligned} f(x,\omega) &= f^0(x,\omega) + \int_V S(x-x')(\mathrm{L}^1F(\mathrm{y}') \\ &+ \rho_1\omega^2\mathbf{g}(R)\mathrm{T}_1 f(\mathrm{y}')]\mathrm{S}(\mathrm{y}')\mathrm{dy}' \end{aligned} \quad (14)$$

In addition, we can introduce the abbreviated formula:

$$g(x,\omega) = \begin{Vmatrix} G_{ik}(x,\omega) & \gamma_i(x,\omega) \\ \gamma_k(x,\omega) & g(x,\omega) \end{Vmatrix},$$

$$S(x,\omega) = \begin{Vmatrix} G_{ik,l}(x,\omega) & \gamma_{i,k}(x,\omega) \\ \gamma_{k,l}(x,\omega) & g_{,k}(x,\omega) \end{Vmatrix},$$

$$L^1(x,\omega) = \begin{Vmatrix} C_{ijkl}^1 & e_{kij}^1 \\ e_{kij}^{1T} & -\eta_{ik}^1 \end{Vmatrix},$$

$$F(x,\omega) = \begin{Vmatrix} u_{(i,j)}(x,\omega) \\ \varphi_{,i}(x,\omega) \end{Vmatrix} \quad (15)$$

In these expression, $G_{ik}(x,\omega)$, $\gamma_i(x,\omega)$, $g(x,\omega)$ can be represented as:

$$g(x,\omega) = \frac{1}{(2\pi)^3}\int g(k,\omega)\exp(-i\mathrm{k}\Box\mathrm{x})d\mathrm{k},$$

$$g(k,\omega) = \begin{Vmatrix} G_{ik}(k,\omega) & \gamma_i(k,\omega) \\ \gamma_k^T(k,\omega) & g(k,\omega) \end{Vmatrix}$$

$$G_{ik}=(\Lambda_{ik}+\frac{1}{\lambda}h_i h_k^T)^{-1},\quad g=-(\lambda+h_i^T\Lambda_{ij}^{-1}h_j)^{-1},$$

$$\gamma_i=\frac{1}{\lambda}h_k^T G_{ki},$$

$$\Lambda_{ik}(k,\omega)=k_j C_{ijkl}^0 k_k-\rho_0\omega^2\delta_{il},\quad h_i(k)=e_{kil}^0 k_k k_l$$

$$,\quad h_l^T=e_{ikl}^{0T}k_i k_k,\quad \lambda(k)=\eta_{ik}^0 k_i k_k$$

$F(x,\omega)$ has nothing to do with coordinate $x_3$. In view of the following relationship

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}e^{-ik_3x_3'}dx_3'=\delta(k_3) \tag{16}$$

Equation (8) can be converted into the following form:

$$f(y,\omega)=f^0(y,\omega)+\int_s S(y-y',\omega)\mathrm{L}^1F(\mathrm{y}',\omega)\mathrm{d}y'$$
$$+\rho_1\omega^2\int_s \mathbf{g}(y-y',\omega)\mathrm{J}f(\mathrm{y}',\omega)\mathrm{d}y' \tag{17}$$

In which, S is cylinder cross section, $y=(x_1,x_2)$ , and

$$\mathbf{g}(y-y',\omega)=\frac{1}{(2\pi)^2}\int_0^{\infty}\bar{k}d\bar{k}$$
$$\int_0^{2\pi}\mathbf{g}(\bar{k},\omega)\exp(-i\mathrm{k}\square(\mathrm{y}-\mathrm{y}'))d\phi\quad,\quad \bar{k}=(k_1,k_2) \tag{18}$$

For such kind of PSO, general form of Equation (13) is:

$$G_{ik}(\bar{k},\omega)=\frac{1}{\rho_0\omega^2}\left[\begin{array}{l}\frac{\beta^2}{\bar{k}^2-\beta^2}\theta_{ik}+\bar{k}_i\bar{k}_k\begin{pmatrix}\frac{1}{\bar{k}^2-\alpha^2}\\-\frac{1}{\bar{k}^2-\beta^2}\end{pmatrix}\\+m_i m_k\frac{\beta_\perp^2}{\bar{k}^2-\beta_\perp^2}\end{array}\right]$$

$$g_{ik}(\bar{k},\omega)=-\frac{1}{\eta_{11}^0}\frac{1}{\bar{k}^2}+\frac{1}{\rho_0\omega^2}\left(\frac{e_{15}^0}{\eta_{11}^0}\right)^2\frac{\beta_\perp^2}{\bar{k}^2-\beta_\perp^2},$$

$$\gamma_i(\bar{k}_i,\omega)=\frac{1}{\rho_0\omega^2}\left(\frac{e_{15}^0}{\eta_{11}^0}\right)^2\frac{\beta_\perp^2}{\bar{k}^2-\beta_\perp^2}m_i \tag{19}$$

In which,

$$\alpha^2=\frac{\rho_0\omega^2}{C_{11}^0},\quad \alpha^2=\frac{\rho_0\omega^2}{C_{66}^0},\quad \beta_\perp^2=\frac{\rho_0\omega^2}{C_{44}'},$$

$$C_{44}'=C_{44}^0+\frac{(e_{15}^0)^2}{\eta_{11}^0} \tag{20}$$

Then we have:

$$\left[\frac{1}{c^2}(\frac{\partial}{\partial t}+\varepsilon)^2-\Delta\right]\bar{g}(r,t)=\delta(t)\delta^2(\mathrm{r}) \tag{21}$$

Another Green function is defined as:

$$\left[\frac{1}{c^2}(\frac{\partial}{\partial t}+\varepsilon)^2-\Delta\right](\frac{\partial}{\partial t}+\varepsilon)^2\bar{h}(r,t)=\delta(t)\delta^2(\mathrm{r}) \tag{22}$$

$$\bar{h}(r,t)=\int_{-\infty}^{\infty}f(t-\tau)\bar{g}(r,\tau)d\tau \tag{23}$$

$f(t)$ is defined as:

$$[\frac{\partial}{\partial t}+\varepsilon]^2 f(t)=\delta(t) \tag{24}$$

$$\bar{g}(k,t)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}\frac{e^{-i\omega t}d\omega}{k^2+(\varepsilon-i\frac{\omega}{c})^2}$$
$$=c^2\Theta(t)\frac{\sin(ckt)}{ck}e^{-\varepsilon t} \tag{25}$$

$$\bar{g}(r,t)=\frac{1}{(2\pi)^2}\int e^{-i\mathrm{kr}}\bar{g}(\mathrm{k},t)d^2k \tag{26}$$

(26) can be converted into:

$$\bar{g}(r,t)=\Theta(t)\frac{c}{(2\pi)^2}\int_0^{2\pi}d\varphi$$
$$\int_0^{\infty}\sin(k[ct-kr\cos\varphi])dk \tag{27}$$

In the equation, the following is adopted:

$$\int_0^{2\pi}\sin(kr\cos\varphi)d\varphi=0 \tag{28}$$

We can get:

$$\int_0^{\infty}\sin k\lambda dk=\lim_{\varepsilon\to0+}\int_0^{\infty}e^{-\varepsilon k}\sin k\lambda dk$$
$$=\lim_{\varepsilon\to0+}\mathrm{Re}\frac{1}{\lambda+i\varepsilon} \tag{29}$$

## IV. EXPERIMENTAL RESULTS

In order to illustrate the actual effect after using particle swarm optimization based on fractal theory in optimization for computer animation trajectory algorithm of trajectory planning control, we choose ten sets of typical index which is shown in Table 1.

TABLE I TEN SETS OF TYPICAL ECONOMY INDEX

| No. | index | weight |
|---|---|---|
| 1 | $K_I$ | 25 |
| 2 | $N_R$ | 10 |
| 3 | $N_{OV}$ | 10 |
| 4 | $E_0$ | 5 |
| 5 | $E_1$ | 5 |
| 6 | $E_2$ | 5 |
| 7 | $P_I$ | 15 |
| 8 | $K_P$ | 10 |
| 9 | $W_S$ | 8 |
| 10 | $K_{PR}$ | 7 |

The comparison before and after using particle swarm optimization based on fractal theory in optimization for computer animation trajectory algorithm of trajectory planning control, ten sets of typical index can be seen from figure 2. The result shows that in the same experimental time, after using the particle swarm optimization based on fractal theory it can achieve better performance in calculating time than before using it.

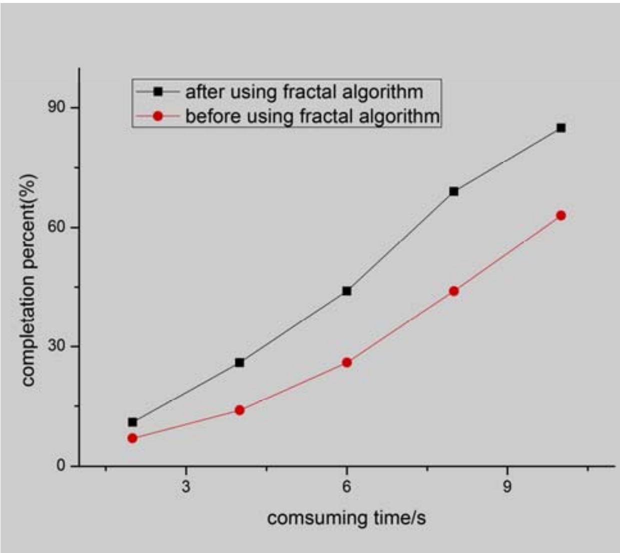
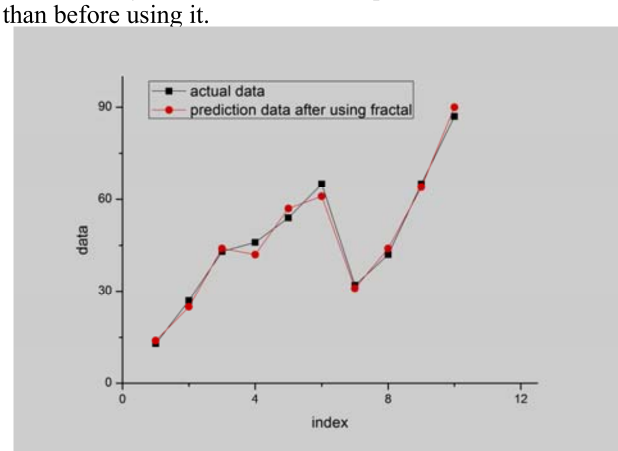



Figure 2. The comparison before and after using fractal algorithm in calculating time.

The research of the optimization for computer animation trajectory and resource allocation is few, and the existing optimization for computer animation trajectory algorithms for trajectory planning control usually laid their attention on the pursuit of the shortest completion time, however, they are not well to take into account the cost of all the tasks for the pursuit of the shortest completion time. The cost of all the tasks is an important factor which cannot be ignored. Different computing resources in cloud platform have different costs; the use of excellent computing resources is higher, while the use of common resources is lower. The cloud users usually make their choices based on comprehensive consideration of the economic budget as well as the wait time. To solve the problem, a double fitness particle swarm optimization algorithm (DFPSO) based on the time and cost constraints are proposed in this paper. The accuracy comparison before and after using particle swarm optimization based on fractal theory in optimization for computer animation trajectory algorithm of trajectory planning control, ten sets of typical index can be seen from figure 3. The result shows that in the same experimental time, after using the particle swarm optimization based on fractal theory it can achieve better performance in accuracy than before using it.



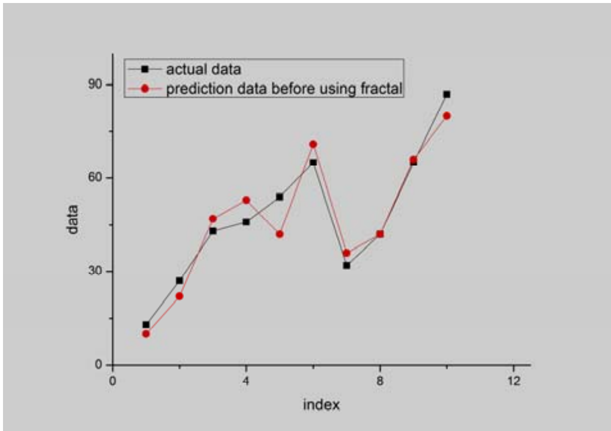




Figure 3. The comparison before and after using fractal algorithm in accuracy.

To a large extent, optimization for computer animation trajectory algorithm affects the performance of the trajectory planning control platform, so improving the optimization for computer animation trajectory algorithm becomes a research hotspot. The research of the optimization for computer animation trajectory and resource allocation is few, and the existing optimization for computer animation trajectory algorithms for trajectory planning control usually laid their attention on the pursuit of the shortest completion time, however, they are not well to take into account the cost of all the tasks for the pursuit of the shortest completion time. The cost of all the tasks is an important factor which cannot be ignored. Different computing resources in cloud platform have different costs; the use of excellent computing resources is higher, while the use of common resources is lower. The cloud users usually make their choices based on comprehensive consideration of the economic budget as well as the wait time.

In the algorithm, both the complete time of all tasks and the cost in computer animation trajectory planning control are used as scheduling objectives. Through this algorithm, the proposed optimization method not only shortens total task completion time and also costs less.

## V. CONCLUSION

The particle swarm optimization (PSO) algorithm is an evolutionary computation technique, which uses the velocity-displacement model through iteration to simulate swarm intelligence. The algorithm initialized with a group of random particles in the space of d dimensions and each particle representing a potential solution is assigned a randomized velocity to change its position to search for the optimal solution. In each iteration, the particles keeps track of the local best solution p best and the global best solution to decide the flight speed and the distance accordingly.

Based on the analysis of the C1oudStack internal resource scheduling mechanism, we find that the two layers of the resources scheduling plays an important role in the task optimal span and load balancing. For the first layer, the virtual machine to the physical machine deployment strategy determines the resource utilization and load imbalance; while for the second layer, the task to the virtual machine allocation strategy determines the optimal span of the task execution time. In this paper, we use the Particle Swarm Optimization (PSO) to optimize the resource scheduling of the two layers. Because the particle swarm algorithm has the advantage of high precision and fast convergence speed, it can fulfill the requirements of the scheduling and shorten operation time, so it is the ideal scheduling algorithm in trajectory planning control environment. To deal with the problem of PSO's premature, Simulated Annealing (SA) is used to optimize PSO. Because the optimization in computer animation trajectory planning control is an important factor which cannot be ignored, different optimization method in control has different costs; the use of excellent computing resources is higher, while the use of common resources is lower. The users usually make their choices based on comprehensive consideration of the economic budget as well as the wait time. To solve the problem, a double fitness particle swarm optimization algorithm (DFPSO) based on the time and cost constraints are proposed in this paper. In the algorithm, both the complete time of all tasks and the cost in computer animation trajectory planning control are used as scheduling objectives. Through this algorithm, the proposed optimization method not only shortens total task completion time and also costs less.

## REFERENCES

[1] Niroshinie Fernando,Seng W. Loke,Wenny Rahayu, “Mobile cloud computing: A survey”. Future Generation Computer Systems,2, 91-95, 2013.

[2] Daniel R. Williams, Peter Thomond, Ian Mackenzie, “The greenhouse gas abatement potential of enterprise cloud computing”. Environmental Modelling and Software, 2:77-89, 2013.

[3] Shih-Wei Lin, Kuo-Ching Ying, “Minimizing shifts for personnel task scheduling problems: A three-phase algorithm”. European Journal of Operational Research, 5: 82-90, 2014.

[4] Reza Salimi, Homayun Motameni, Hesam Omranpour, “Task scheduling using NSGA II with fuzzy adaptive operators for computational grids”. Journal of Parallel and Distributed Computing, 2014.

[5] Yuming Xu,Kenli Li,Jingtong Hu,Keqin Li, “A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues”. Information Sciences, 15: 112-118, 2014.

[6] M.G. Avram, “Advantages and Challenges of Adopting Cloud Computing from an Enterprise Perspective”. Procedia Technology,12: 298-306, 2014.

[7] Reza Rezaei, Thiam Kian Chiew, Sai Peck Lee, Zeinab Shams Aliee, “A semantic interoperability framework for software as a service systems in cloud computing environments”. Expert Systems With Applications,4: 113-119, 2014.

[8] Younis A. Younis, Kashif Kifayat, Madjid Merabti, “An access control model for cloud computing”. Journal of Information Security and Applications, 7: 22-33, 2014.

[9] Yizhuo Wang,Yang Zhang,Yan Su,Xiaojun Wang, Xu Chen, Weixing Ji, Feng Shi, “An adaptive and hierarchical task scheduling scheme for multi-core clusters”. Parallel Computing, 10: 909-911, 2014.