

## Research on Task Scheduling Convergence Non-Dominated Sorting Method in Cloud Computing

Liang. Huang, Jun. Zheng and Guolv.Tan

*School of Mathematics & Computer Science, Shangrao Normal University,  
Shangrao, Jiangxi, China  
E-mail:light\_h@163.com*

### **Abstract**

*Cloud computing task scheduling is a multi-objective decision on how to perform a variety of tasks simultaneously reasonable sort is essential. Therefore, this article constructs a population of convergence non-dominated sorting method. This method is based on non-dominated sorting method; the use of distributed estimation method is improved by four steps to complete the task order scheduling. The experimental results show that this method has good convergence and distribution.*

**Keywords:** *Cloud Computing; Population convergence; Non-dominated sorting, Distribution..*

### **1. Introduction**

Due to scheduling policies have a significant impact on the performance of cloud computing, so in a heterogeneous environment, such as a grid, research has been done on the issue of workflow task assignment and scheduling. Before a lot of work focuses on the use of heuristic-driven scheduling algorithm to complete the workflow [1]. However, these algorithms can not be applied in the cloud computing environment, because the resources they needed to be limited. Therefore, in recent years, workflow scheduling algorithms in the cloud began to spread. on Scheduling algorithms are classified in reference[2], considering local information task level scheduling algorithm is based on an independent task scheduling strategy decisions, workflow-level scheduling algorithm is the entire workflow as a whole into account and the implementation schedule. In reference [3], the authors used a Markov decision process, based on the execution of the external environment, to guide the workflow; in addition, the system uses a genetic algorithm to evolve workflow scheduling. If the implementation of the provisions of the corresponding task scheduling constraints of time, which can be refined to scheduling constraints perform each task, then the deadline for each task based on the task and resource mapping, optimizing the total implementation costs, which there are many kinds of heuristics: such as time-based allocation of TD (Time Distribution)algorithm share in each task length approach path allocation based on the proportion of time[4], reverse stratification algorithm DBL of deadline constraints (Deadline Bottom Level) Forward stratification algorithm of deadline constraints DTL (Deadline Top Level), respectively, task-based and forward depth on the task were stratified according to time allocation layer, the task in the allotted time cost optimization[5].

In reference [6], the authors designed a multi-objective integrated ant colony optimization scheduling algorithm. In order to measure the uncertainty, the concept of entropy is used to update the pheromone of ant colony algorithm, an improved algorithm to accelerate the convergence rate of the solution set. In order to make the scheduled time to a minimum, the introduction of the minimum Min-min algorithm to calculate the optimal time limit to complete the task scheduling as heuristic information. For scientific

workflow in environments heterogeneous multi-objective design problems of static scheduling a general framework and heuristics [7]. The algorithm for each target uses user-specified of constraints by applying a double strategy approximation optimal solution. On the impact of different objectives for optimizing processes were analyzed and categorized, and presented a case of four goals, including the completion time, the economic costs, energy consumption and reliability. In reference [8], the authors propose a model based on particle swarm optimization for mapping between tasks and resources, in order to achieve within the deadline specified by the user, so that the task consumes minimal.

Through literature review of existing research results can be seen, task scheduling cloud computing is a typical multi-objective optimization problem. In this paper, non-dominated sorting algorithms and estimation of distribution algorithms together, presents a population of convergent non-dominated sorting task scheduling approach to enhancing the performance of cloud computing task scheduling.

## 2. Populations of Convergence Non-dominated Sorting Method

In this paper, the population of non-dominated sorting method convergence, built on the basis of estimation of distribution algorithms and non-dominated sorting method above, and therefore the first to analyze these two methods.

### 2.1 Non-dominated Sorting Method

Fast non-dominated sorting algorithm for population stratification, and the use of crowding in the same layer as the standard of excellence, which makes the Pareto optimal solutions distributed more evenly, while maintaining the diversity of the population. In order to expand the sample space, the progeny population and its parent populations were combined, and then allowed to compete with each other to produce the next generation of the population, the so-called elitist strategy. With this strategy can not only make excellent individual in to the next generation from the parent population, but also improve the overall level of population, because the individual hierarchical storage reasons, the best individual in favor of heritage.

#### (a) Fast non-dominated sorting algorithm

Assuming the size of the population (Pop) is  $N$ , wherein each individual (i) have two parameters control the population of individuals for Solution i, which is defined as the number of  $n_i$ ; i individual solutions dominated solutions in the population, which is defined as a set of  $S_i$ . Pop the population stratification is divided into m layers each assuming a subset of the total of m subsets  $F_1, \dots, F_m$ , then  $\bigcup F_j = Pop$ , where  $j = 1, 2 \dots m$ . Algorithm as follows:

(1) The collection is stored in  $F_1$ ,  $N = 0$ , individual solutions.

(2) Through the collection of individual  $F_1$  in solution, a solution for every individual dominating set of solutions of individual  $S_j$  k, its value is  $n_k$  minus 1, then  $n_k = 0$  into a collection of individuals in  $F_2$ .

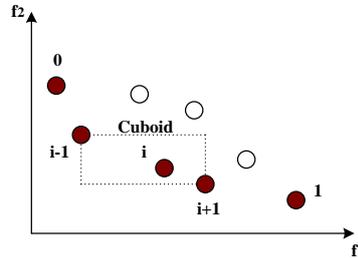
(3) The first layer of non-dominated among individuals, that is, an individual step in the solution of (1) a set of  $F_1$ .

(4) The collection as a collection repeat the above steps (1)(2)(3) of the current. When individuals in the population (Pop) are good points layer, the algorithm will stop.

#### (b) Crowding

When a new generation of populations if individuals gathered to adapt to higher value less dense, it will be more evenly distributed, there is a greater chance of being retained,

then the population as a parent, to participate in the next generation of genetic them go. The smaller the individual packing density solution, then the greater will be gathered from this distance that is crowding the individual  $P[i]_{distance}$ . First, to calculate the objective function value  $Pop(t)$  of the solution of the individual, and the function values depending on the size of the individual  $i$  sort solution, the final calculation of the degree of congestion of individual  $i$ . For dual-objective optimization is concerned, shown in Figure 1, crowding the solution of individual  $i$  that is individual  $i+1$  and the individual  $i-1$  for the vertices of the rectangle and the length of sum.



**Figure 1. Pair of Goals in the Solution of Individual  $i$  Crowding**

(1) Determine the degree of congestion

Shown in Figure 1, there are two objective functions  $f_1$  and  $f_2$ , assuming the solution the congestion degree in the population of individuals  $Pop$   $i$  is  $P[i]_{distance}$ , the individual function value  $i$  on the objective function  $f$  is set to  $P[i].f$ , you can launch crowding of individual  $i$  is:

$$P[i]_{distance} = (P[i+1].f_1 - P[i-1].f_1) + (P[i+1].f_2 - P[i-1].f_2) \quad (1)$$

Under normal circumstances, assuming that the objective function with  $R$ , the formula is calculated the degree of congestion of an individual:

$$P[i]_{distance} = \sum_{k=1}^R (P[i+1].f_k - P[i-1].f_k) \quad (2)$$

(2) Crowding comparison operator

By sorting and crowding relatively quick get on two parameters of the population of each individual solution:  $i_{rank}$  and  $P[i]_{distance}$ , For individual  $I$  is non dominated solution layer level and degree of congestion. Definition of crowding comparison operator  $\succ_n$  to:

$$i \succ_n j \quad \text{if } (i_{rank} < j_{rank} \text{ or } i_{rank} = j_{rank} \text{ and } P[i]_{distance} > P[j]_{distance}) \quad (3)$$

When this solution  $i$  is superior to the individual solutions  $j$ , i.e., when the type of solution individual is not the same non-dominated layers in order to take a small number of non-dominated layer solution can be selected to ensure the optimal solution; When different individuals in the same solution of non-dominated layers, take crowding larger individuals, so that the solution is more evenly distributed. Comparing the congestion degree of the operator to ensure the population  $Pop(t)$  of the diversity of the population so that a uniform solution of the individual to a Pareto distribution surface.

(c) Genetic manipulation

Breeding populations of non-dominated sorting algorithm to generate the use of genetic algorithms, in order to get the optimal solution, genetic algorithms must be

designed to meet the purpose of the experiment, which also reflects the importance of the genetic algorithm. In the design of the algorithm, the genetic algorithm is based on the nature of the design of the real situation biological reproduction, including the selection, crossover and mutation.

Non-dominated sorting algorithm is used to select the individual binary tournament selection, crossover and mutation operations and is mainly used simulated binary crossover and polynomial mutation.

(1) Simulated binary crossover SBX

SBX crossover operator is used up, Equation (4):  $p_{1,k}$  and  $p_{2,k}$  for the parent on behalf of individuals in the population,  $c_{1,k}$  and  $c_{2,k}$  represent the individual child.

$$\begin{aligned} c_{1,k} &= 0.5 * [(1 - \beta_k) p_{1,k} + (1 + \beta_k) p_{2,k}] \\ c_{2,k} &= 0.5 * [(1 + \beta_k) p_{1,k} + (1 - \beta_k) p_{2,k}] \end{aligned} \quad (4)$$

Here,  $c_{i,k}$  expresses  $i$  of progeny,  $p_{i,k}$  represents the selected parent,  $\beta_k$  is greater than or equal to 0, the pseudo-random generation of random variables, Generated as follows:

$$\begin{aligned} p(\beta) &= 0.5 * (\eta_c + 1) \beta^{\eta_c} & 0 \leq \beta \leq 1 \\ p(\beta) &= 0.5 * (\eta_c + 1) \frac{1}{\beta^{\eta_c} + 2} & \beta > 1 \end{aligned} \quad (5)$$

Where,  $\beta$  is a random variable, we need to regenerate on every dimension, let  $u \in (0, 1)$ , randomly generated as shown in equation (6).

$$\begin{aligned} \beta(u) &= (2u)^{\frac{1}{\eta+1}} \\ \beta(u) &= \frac{1}{(2(1-u))^{\frac{1}{\eta+1}}} \end{aligned} \quad (6)$$

(2) Polynomial mutation

$$c_k = p_k + (p_k^u - p_k^l) \delta_k \quad (7)$$

Expressed progeny  $c_k$ ,  $p_k$  represents the parent,  $p_k^u$  represents the boundary of the parent composition,  $p_k^l$  represents the lower bound,  $\delta_k$  and represents a very small number.

**2.2 Distribution Estimation Method**

This article is selected univariate marginal distribution estimation algorithm, the univariate marginal distribution probability vector estimation algorithm changes by the following formula:

$$p(x, t+1) = \prod_{i=1}^n p(x_i, t+1) = \prod_{i=1}^n \frac{\sum_{j=1}^n \delta_j (X_i = x_i)}{u} \quad (8)$$

Here is a univariate marginal distribution estimation algorithms specific example. The examples assume that the goal is to find the maximum value of the function, that is  $\max\{f(x) = \sum_{i=1}^n x_i\}$ , where  $x \in \{0,1\}$ , assuming probability vector  $p = (p_1, p_2, \dots, p_n)$  with  $n = 3$ , this is a description of the solution space probability model.  $Pop(t)$

represents the probability distribution  $p$ , Each gene  $i=1$  corresponds to the probability  $p_i$ . Corresponding,  $i = 0$  corresponds to a probability  $1 - p_i$ .

The first step, initial population Pop(0). Pop(0) individual solutions in a manner distributed uniformly randomly generated in the solution space. Let  $p = (0.5, 0.5, 0.5)$ . Initial population Pop (0) has eight individual solutions. By fitness

$f(x) = \sum_{i=1}^n x_i$  calculated population fitness value of each individual solution, as shown in

Table 1.

**Table1. The First-generation Group**

	x1	x2	x3	f(x)
1	0	0	1	1
2	1	1	0	2
3	0	0	0	0
4	0	1	1	2
5	0	1	0	1
6	1	0	0	1
7	1	0	1	2
8	1	1	1	3

**Table 2. Clod and Heat Selected based on the Fitness Advantages of Groups**

	x1	x2	x3	f(x)
2	1	1	0	2
4	0	1	1	2
7	1	0	1	2
8	1	1	1	3

The second step: compare the fitness values; because it is seeking the maximum operation, so elected four relatively high fitness value individual solutions in the population Pop (0), they are used to update the probability vector  $p$ , as shown in Table 2.  $X_s$  represents advantaged groups of four solutions consisting of elected individuals. To update the probability vector  $p$  by the formula  $p_i = P(x_i = 1 / X_s)$  :

$$p_1 = P(x_1 = 1 / X_s) = p_2 = p_3 = 0.75$$

The third step: Updated probability vector  $p$ , is generated a new individual and based on  $p$ , combined into anew population Pop(t).  $p$  is a description of the distribution of each possible solution, Assuming the solution  $b = (b_1, b_2, \dots, b_n)$ , the probability of individual B is:

$$P(b) = P(x_1 = b_1, x_2 = b_2, \dots, x_n = b_n) = \prod_{i=1}^n P(x_i = b_i) = \prod_{i=1}^n |1 - b_i - p_i| \quad (9)$$

For example,  $b = (1, 1, 0)$ , then  $P(1,1,0) = 0.75 * 0.75 * 0.25 \approx 0.14$ . Then, in order to generate new individuals, the model is used in performing the sampling operation. So far is a periodic distribution estimation algorithms, return to the second Step, From the new generation groups choose excellent individuals, updated probability model  $p = (1.00, 0.75, 0.75)$ , then random sampling, production of new individuals. In this case, the value of each gene are to obtain the optimal solution 1.00, with the continuous implementation of the estimation of distribution, the probability distribution of the optimal solution with the evolution of the population, followed by 0.075, 0.14, 0.56. So the

probability of occurrence of individual optimal solution is sent to be seen in the constant change.

### 2.3. Distribution Estimation Method

In this paper, the algorithm will univariate marginal distribution estimation algorithm and non-dominated sorting method combining full use of their respective advantages, beginning at the time of execution of the algorithm, due to the solution of the initial population of individuals scattered with in the relationship between them is relatively small, so I chose a non-dominated sorting genetic manipulation method (crossover and mutation) to generate the probability of an individual than the use of univariate marginal probability distribution model estimation algorithm to generate new individuals to a large probability. With the execution of the algorithm until convergence in distribution, population to a predetermined expectations, population of individual starts to present the distribution of solutions of certain when the estimation algorithm model to produce the probability solution of the individual becomes larger with the univariate marginal distribution.

Genetic algorithms commonly referred to as convergence, simply refers to the process of reproduction from generation to generation, the fitness value of the population is not significantly improved. In the course of the continuous generation of offspring, and individual solutions continue to move closer to the Pareto front, showing continued convergence trend. Individual solutions according to certain rules distribution in the target space in this section to determine the population convergence approach is not the ultimate convergence of the population of an assessment, but to generate new populations of which method to the solution according to the degree of convergence of individual judgment in the individual.

Target space solution set is composed of one point, so this problem can be transformed into point set convergence judgment. Steps are as follows:

(1). Determine the geometric center point set. In the population  $Pop(t)$ , respectively based on the two objective function calculate the average of all individuals in the population, i.e., the objective function is the average value ( $m_t^1$  and  $m_t^2$ ) of  $f_1(x)$  and  $f_2(x)$ . So that coordinates of point  $m_t$  ( $m_t^1, m_t^2$ ).

(2). Calculation of the target space all point to the geometric center distance. Population  $Pop(t)$  the solution of the individual  $X_j(f_{1t}^j, f_{2t}^j)$  and  $X_i(f_{1t}^i, f_{2t}^i)$ , the distance between points is calculated as:

$$d_{i,j} = \sqrt{(f_{1t}^j - f_{1t}^i)^2 + (f_{2t}^j - f_{2t}^i)^2} \quad (10)$$

(3). Calculate the distance of the variance and standard deviation.  $E(d_t)$  mean value represents the average mean distance of the length,  $D(d_t)$  variance represents the Variable to the degree of deviation from mean. Standard deviation  $\sigma(d_t) = \sqrt{D(d_t)}$  with the same dimensionless variables, therefore, by comparing the standard deviation to determine whether the convergence of the predetermined desired.

(4). Criterion. The t generation population individual distance of standard deviation if there is no obvious change and the average standard deviation of three generation is that the convergence of population distribution conforms to a predetermined desired, that is distribution has certain regularity. This increases risk of population individuals through the probability model of sampling.

$$\left| \sigma(d_t) - \frac{1}{3} \sum_{i=1}^3 \sigma(d_{t-1}) \right| < \varepsilon \quad (11)$$

### 3. Experiment and Analysis

The In order to verify the performance of the non-dominated sorting method converges in population paper constructs the use of benchmarking functions for verification. The function used herein includes three features experiments.

- (1)Present a convex or non-convex Pareto frontier.
- (2)Search space with offset.
- (3)Pareto frontier is discrete or continuous.

These three features make the solution of individual populations closer to the true Pareto front of the difficulty increases, but also makes the non-inferior solution is not easy to maintain the diversity of characteristics. Were given three validation function, ZDT 1as shown in equation(12), it is a continuous function and is convex Pareto frontier; ZDT 2 as shown in equation(13), because it contains the sine function, so the Pareto front is discrete.

$$ZDT1: \begin{cases} M \inf_1 = x_1 & g = 1 + 9 \cdot \sum_{i=2}^n x_i / (n-1) \\ M \inf_2 = g \cdot h & h = 1 - \sqrt{f_1 - g} \end{cases} \quad (12)$$

Among them, the number of variables  $n = 30$ , Pareto frontier ZDT 1is convex curve.

$$ZDT2: \begin{cases} M \inf_1 = x_1 & g = 1 + 9 \cdot \sum_{i=2}^n x_i / (n-1) \\ M \inf_2 = g \cdot h & h = 1 - (f_1 / g)^2 \end{cases} \quad (13)$$

Among them, the number of variables  $n = 30$ , Pareto frontier ZDT 2 is non-convex curve.

Measure the following criteria:

- (1)Convergence measure  $M_1$ .

$$M_1(Y^l) = \frac{1}{|Y^l|} \sum_{b^l \in Y^l} \min \{ \|b^l - \bar{b}\|; \bar{b} \in \bar{Y} \} \quad (14)$$

In equation (15),  $Y^l$  is a set of target vectors, with the non inferior solutions in the process of population evolution solution individuals generated solution corresponding to the set.  $\bar{Y}$  is a set of target vectors too , it and non-inferiority generated by the algorithm theory generate to correspond.  $M_1$  collection embodies the target vector  $Y^l$  values in the shortest distance to the mean value of  $\bar{Y}$ .  $M_1$  the smaller, illustrates actual results and the theoretical solution of the smaller the distance, corresponding to the better performance of its convergence.

(2)Distribution of the measure (sd): Individual solutions or the target spatial distribution algorithm to measure the quality of a certain role. Standard deviation of the distribution of the situation described. Measure sd equation(16) shown in.

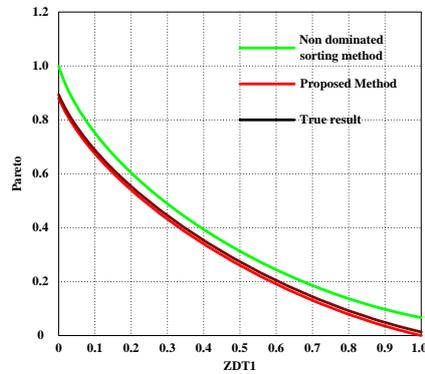
$$sd = \sqrt{\frac{\sum_{i=1}^n (d_i - \bar{d})^2}{n}} \quad (15)$$

Here,  $n$  is the number of individuals in the non dominated solutions in the population.  $d_i$  represents the distance and represents the minimum distance from one individual to the solution of other solutions.  $\bar{d}$  is the average of the shortest distance  $d_i$ .sd smaller, which means that the distribution of solutions for the individual, the better. When  $sd = 0$ , the non inferior solution of each individual in the objective space appear equidistant distribution in population.

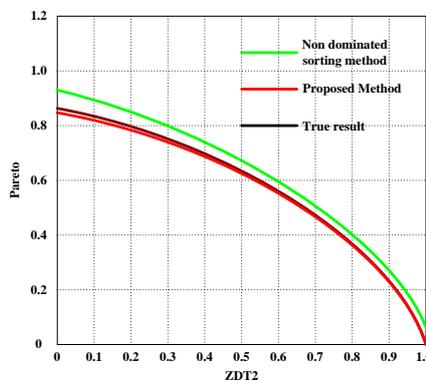
Figure 2 and Figure 3 shows the comparative results of Pareto frontier this improved method and non-dominated sorting method and are based ZDT1 test functions and test functions ZDT2.

As can be seen from the results in Figures 2 and 3, in the test functions ZDT1 and ZDT2, in this paper, an improved method of Pareto frontier and actual results closer than the non-dominated sorting method, this also explains the validity of this improved method.

Further comparison of the two methods of the convergence of  $M_1$  and distribution of the sd results, as shown in Table 3.



**Figure 2. ZDT1 Pareto Front Comparison Results**



**Figure 3 ZDT2 Pareto Front Comparison Results**

**Table3. Comparison of Non-dominated Sorting Method and the Proposed Method**

		ZDT1	ZDT2
$M_1$	Non-dominated method	0.0206	0.1997
	Proposed method	0.0072	0.0065
sd	Non-dominated method	0.0439	0.0299
	Proposed method	0.0117	0.0115

As can be seen from Table3, in this paper, through the improvement of this method, cloud computing tasks scheduling for processing performance has been significantly improved. This method as compared with non-dominated sorting method, convergence performance greatly improved, distribution sd almost close to zero.

#### 4. Conclusion

In order to improve the performance of task scheduling in cloud computing, in this paper, non-dominated sorting method and univariate marginal distribution estimation method carried out in-depth study. Finally the two ways together, constructed a population convergence of non-dominated sorting method. This method is obtained by taking the geometric center point set task, a target space from the geometric center point of the calculated distance of variance and the standard deviation, exercise population iteration criterion scheduling four steps. Experimental results show that this method than the general non dominated sorting method, has been significantly improved in the convergence and diversity index.

#### Acknowledgements

This work is supported by the Scientific Research Fund of Jiangxi Provincial Education Department of China (Grant No.GJJ13704)

#### References

- [1] M. Perez, M. Juan and J. B. Bernable, "Taxonomy of trust relationships in authorization domains for cloud computing", *Journal of Supercomputing*, (2014), pp.1-25.
- [2] J. Park, H. Yu, H. Kim and E. Lee, "Dynamic group-based fault tolerance technique for reliable resource management in mobile cloud computing", *Concurrency Computation Practice and Experience*, (2014).
- [3] A. Gele, "Study on the security of the databases in cloud computing environment", *Applied Mechanics and Materials*, (2014), pp.473-476.
- [4] V. Badesou and A. Dumitresou, "Simple models to compute solar global irradiance from the CMSAF product Cloud Fractional Coverage", *Renewable Energy*, vol. 66, (2014), pp.118-131.
- [5] G. Martinez, S. Zeadally and H. C. Chao, "Editorial: cloud computing service and architecture models", *Information Sciences*, vol. 258, (2014), pp. 353-354.
- [6] F. Belqasmi and C. Azar, "A case study on IVR applications provisioning as cloud computing services", *IEEE Network*, vol. 28, no. 1, (2014), pp. 33-41.
- [7] S. A. Odunaike and O. O. Olugbara, "Mitigating rural e-learning sustainability challeges using cloud computing technology", *Lecture Notes in Electrical Engineering*, vol. 247, (2012), pp. 497-511.
- [8] J. W. Lin, C. H. Chen and C. Y. Lin, "Integrating QoS awareness with virtualization in cloud computing systems for delay-sensitive applications", *Future Generation Computer Systems*, (2014).

#### Authors



**Liang. Huang.** Associate Professor, School of mathematics and computer science in Shangrao Normal University. Born in 1981, Mr Huang graduated and got master degree from Huazhong University of Science and Technology, and his main research interests are Internet Safety and Cloud Computing.



**Jun. Zheng.** Senior secondary school teachers, The Education Bureau of Shangrao City, Born in 1974, Mr. Zheng served as Xinzhou District Lingxi high school principal Shangrao city of Jiangxi province, main research direction of school education informationization.



**Guolv.Tan.** Professor of the school of mathematics and computer science in Shangrao Normal University, Jiangxi Normal University, master's tutor, Born in 1956,main research direction in data mining.