N-D Normal Membership Cloud Model Based on Region Partition

GOU Jin
College of Computer Science and Technology, Huaqiao University, Quanzhou, China
Email: goujin@hqu.edu.cn

ZHAO Zhi-Qiang, CHEN Wei-Bin, LUO Wei
College of Computer Science and Technology, Huaqiao University, Quanzhou, China
Email: {chenwb, zhiqiang1285, luowei}@hqu.edu.cn

***Abstract*—Uncertainty exists widely in the subjective and objective world. Randomness and fuzziness are the two most important and fundamental forms of them. The cloud model measured by expectation, entropy and hyper-entropy can describe this uncertainty well. A new algorithm of forward and backward cloud based on region partition is proposed. After a comparison was made between these two kinds of algorithms, the conclusion that cloud model based on region partition is better for describing uncertainty than the older one could be drawn.**

***Index Terms*—N-D Normal Cloud; Cloud Model Based on Region Partition; Membership Cloud; Uncertainty**

I. Introduction

During the history of artificial intelligence, the representation of knowledge holds a large part. But in real word, lots of phenomena are uncertain. The certain and inerratic phenomenon occurred after specifically condition, and existed only in a short and local range of time [1]. As a result of random, fuzziness, incompleteness and disagreement of description, the studies on uncertainty are partial in artificial intelligence [2].

The mathematic implements for handing uncertainty mostly are probability theory and fuzzy mathematics. Bayesian theory uses prior probability and samples data to compute and estimate unknown samples. The uncertain inference models by credence putted forward by Shortliff and evidence theory proposed by Dempster are representation of knowledge's random. In 1948, Shannon introduced entropy which appeared in energy information domain. Entropy can be used to describe the average uncertainty of an idiographic uncertain affair which appears in a case set. The instrument of handing fuzziness is fuzzy theory [3]. The membership degree and membership function map a conception to a real number in the interval [0,1]. The numerical value between 0 and 1 can be used to express the membership degree of a conception. Methods such as fuzzy predicate, fuzzy rules and fuzzy framework will be a fuzzy treatment of precise knowledge.

Based on the above ideology, Li De-Yi proposed cloud model theory to describe the uncertainty of knowledge [4]. Membership cloud employs expectation *Ex*, entropy *En* and hyper-entropy *He* to describe a specific concept. *Ex* expresses the point which is the most suitable to represent the domain of the concept and it is the most typical sample after this concept to quantify. *En* represents a granularity of a concept which could be measured (the larger of entropy, the larger of the granularity, the concept is more macro). It reflects the range of domain space which could be accepted by the specific concept. *He* describes the uncertain measurement of entropy. It can be used to express the relationship between randomness and fuzziness [1].

In real word, normal cloud attached lots of people's attention because there is a lot of ambiguity in the concept which uses the normal function to describe is the closet model of human thinking [5]. Cloud model unfolds a good character when it is used to deal with actual knowledge of uncertainty. The usage of similar cloud [6] in measuring and analyzing the uncertainty indicate that cloud model is an ideal tool for copying with the uncertainty of knowledge. Traditional genetic algorithm (GA) easily gets stuck at a local optimum, and often has slow convergent speed. Cloud-model-based genetic algorithm was proposed in [7] which are based on both the idea of GA and the properties of random and stable tendency owned by a normal cloud. It can solve the drawback of traditional GA better. Combing watermark algorithms with cloud model could protect copyright of relational database well [8]. The idea of cloud can also be applied to the space knowledge mining and image processing, and achieved better results [9-11].

In this paper, we first introduce the definition of membership function and membership cloud. Then we implement the generator of forward normal cloud and backward normal cloud. In the fourth section, we propose 2-D cloud model based on region partition and its application. In section five, we analyze the drawback of

This work was supported by National Science Foundation Project (*No: 60872057*), Natural Science Foundation of Fujian Province of china (*No: A071003*);

doi:10.4304/jsw.5.3.288-295

original cloud algorithms and make a comparison between them. Lastly we conclude with a summary and directions for future work in section VII.

II. Membership Function and Membership Cloud

In Knowledge representation, it is more visual and more intuitive to describe knowledge using concept than using accurate mathematical description. This paper uses membership cloud to describe the uncertainty of knowledge. The cloud model can also be seen as a transfer model between the quantitative and qualitative description of knowledge. The Definition of Membership Function, Information Entropy, Membership Cloud Model and Normal Cloud Model will be described in the following section.

A. Membership functions in fuzzy mathematics

Definition 1: Membership Function [2]. Let X denote an ordinarily set, $x \in \{x\}$ which is called a domain. $\tilde{A}$ is the fuzzy subset on the domain X, which is defined: $\forall x \in \{x\}$, there always exists a numerical variable $\mu_{\tilde{A}}(x)$, which belongs to the interval [0,1]. The numerical variable $\mu_{\tilde{A}}(x)$ is called the element x 's membership degree on $\tilde{A}$. Then the mapping:

$$\mu_{\tilde{A}} : x \to [0,1], \forall x \in X, x \to \mu_{\tilde{A}}(x) \qquad (1)$$

is called $\tilde{A}$'s membership function.

Membership function is the basis of fuzzy math. The most common membership functions used in fuzzy mathematics are[12]:

1) Liner membership function: $\mu_{\tilde{A}}(x) = 1 - kx$;

2) Γ membership function: $\mu_{\tilde{A}}(x) = e^{-kx}$;

3) Convex membership function: $\mu_{\tilde{A}}(x) = 1 - ax^k$;

4) Cauchy membership function: $\mu_{\tilde{A}}(x) = \frac{1}{1+kx^2}$

5) Mountain-shaped membership function:

$$\mu_{\tilde{A}}(x) = \frac{1}{2}(1 - \sin(\frac{\pi}{b-a} \cdot (x - \frac{b-a}{2})))$$

6) Bell-shaped membership function: $\mu_{\tilde{A}}(x) = e^{\frac{-(x-a)^2}{2b^2}}$.

The bell-shaped membership function is mostly used in daily life among them. Membership functions are the bridge between qualitative and quantitative description of a concrete concept. A conversion from qualitative to quantitative description can be drawn through the membership degree. But how to select a clear and concrete membership function is one of the complex issues in the study of fuzzy sets.

B. Entropy

Entropy is an important factor while studding uncertain intelligence. It first was introduced into Thermodynamics by R. J. Clausius in 1864 to express the uniformity of any energy in space. Shannon introduced in to the area of Information Science in 1948[13]. The entropy of information can be described as:

Definition 2: Information Entropy Assume a system X is formed by a series of cases X_i, where $X = \{X_i \mid i=1,2,..,n\}$, and the probability of X_i is $p(X_i)$. Then the information entropy can be defined:

$$H(X) = -\sum_{i=1}^{n} p(X_i) \log p(X_i) \qquad (2)$$

Information entropy is used to describe the average uncertainty of appearance of X_i in cases set X.

The larger entropy, the more uncertain for information. If the probability of X_i is equal to each other for case set X, when the information entropy reaches its maximum.

Entropy has widely usages after it was proposed such as in statistical physics and quantum physics areas appears the heat-entropy, electronic-entropy, spin-entropy and so on, and in other area such as topology-entropy, social-entropy, cultural entropy et[14]. Through the history of entropy we can see that the entropy is useful equipment for measuring the uncertainty. Based on its attributes, it was used in cloud model for measuring the granularity of a concept.

C. Membership Cloud & Normal Cloud

Based the description of membership function in fuzzy theory and information entropy for information theory, the Membership cloud can be defined as follows:

Definition 3: Membership Cloud [2]. Let U denote a quantitative domain composed by precise numerical variables; C is a qualitative concept on U. If the quantitative value $x \in U$ is a random realization of qualitative concept C, then the x 's confirmation on C can be denoted $\mu(x) \in [0,1]$, which is a random number with stable tendency.

$$\mu : U \to [0,1], \forall x \in U, x \to \mu(x) \qquad (3)$$

The distribution of x on U is called Cloud, x is called a Cloud Droplet.

The cloud is from a series of cloud drops. In the process of the formation of clouds, a cloud droplet is a realization of qualitative concept through numeric measurement. The realization order between the cloud droplets is irrelevant. The random realization in Def. 3 is under the sense of probability, and the confirmation in Def. 3 is the membership degree of qualitative concept under fuzzy sense. As a single cloud droplet of little or no practical significance, just consider the whole clouds emerged by the characteristics of an individual rather than considering the specific characteristics of the cloud droplet.

On the probability of normal distribution is the most commonly used form of distribution. It is described by expectation E and variance D. In fuzzy set theory, the bell-shape membership function $\mu(x) = e^{\frac{-(x-a)^2}{2b^2}}$ is also the most common membership function used in fuzzy sets. Normal cloud combines the characteristics of two and then makes a further expansion. Normal cloud employs expectation, entropy and hyper-entropy to make the cloud generator, and generate the conversion model of qualitative description and quantitative description of the concrete concept.

Definition 4: Normal Cloud [2]. Let U denote a quantitative domain composed of precise numerical variables; C is a qualitative concept on U. If the quantitative value $x \in U$ is a random realization of qualitative concept C, $x \sim N(Ex, En'^2)$, $En' \sim N(En, He^2)$ and x 's confirmation on C is

$$\mu = e^{-\frac{(x-Ex)^2}{2(En')^2}} \tag{4}$$

Then the distribution of x on U is called Normal Cloud.

As show in Fig. 1, there is a 2-d Normal Cloud Which the expectation is (0,0), entropy is (1,1) and hyper-entropy is(0.05,0.05). This Cloud owns 200 cloud droplets and the vertical direction describes the confirmation of the droplets.

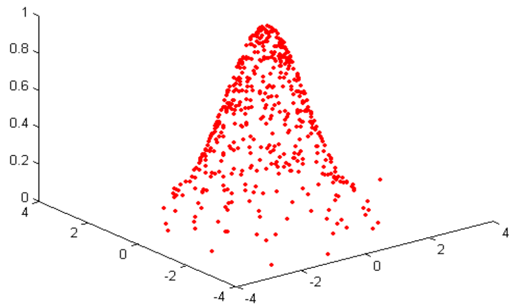

Fig.1 A 2-d Normal Cloud with Ex = (0, 0), En = (1, 1) and He = (0.05, 0.05). There are 200 cloud droplets in this normal cloud.

III. Forward and Backward Normal Cloud Generator

Cloud model employs expectation *Ex*, entropy *En* and hyper-entropy *He* to describe a concrete concept. It is a useful conversion model between the qualitative and quantitative description for the concept. The basic tool for conversion is forward cloud generator and backward cloud generator.

A. *Forward Normal Cloud Generator*

Forward normal cloud generator is the most commonly used in cloud generator. It is a mechanism for mapping from qualitative to quantitative description. The model of forward normal cloud generator is shown in Fig. 2.

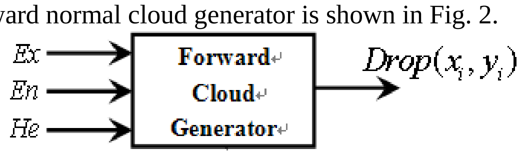

Fig.2 The model of Forward Cloud Generator. It employs Exception *Ex,* entropy *En* and hyper-entropy *He* to generate a series droplets.

By inputting the number of its characteristics, forward normal cloud generator produces cloud droplets described by the specific concept. Forward normal cloud generator uses the law of normal distribution, passes two random processes to generate the quantitative representation of the concrete concept which is called the cloud droplets group. A single cloud droplet has little or no practical significance, so we usually consider only cloud droplet group as a whole state.

The realization of forward normal cloud generator [2] can be described as follows:

Algorithm 1: *ForCloud* (*Ex, En, He, N*)

Input: *Ex, En, He, N*

Output: $Drop(x_i, conf(x_i))$

Step: Execute the following steps

step1: Generate a *m*-dimensional normal random number En' with the expectation *En* and stand deviation *He*, $En' \sim N(En, He^2)$;

step2: Generate a *m*-dimensional normal random number x_i with the expectation *Ex* and stand deviation En', $x_i \sim N(Ex, En'^2)$;

step3: Compute the confirmation of the cloud droplet x_i.

$$conf(x_i) = e^{-\sum_{j=1}^{m}\frac{(x_{ij}-Ex_j)^2}{2En_j'^2}} \tag{5}$$

step4: Generate cloud droplet $Drop(x_i, conf(x_i))$;

step5: Repeat step1-step4, until $i = N$.

In algorithm 1, *Ex* describe expectation, *En* describe entropy, *He* describe hyper-entropy, *N* describe the number of droplets, $Drop(x_i, conf(x_i))$ describe a cloud droplet where x_i describe an *m*-dimensional number and $conf(x_i)$ is the confirmation of x_i.

B. *Backward Normal Cloud Generator*

Backward normal cloud generator can obtain the qualitative description of a concrete concept such as expectation *Ex*, entropy *En* and hyper-entropy *He* from a series quantitative cloud droplets which have been given. The model of backward normal cloud generator is shown in Fig. 3.

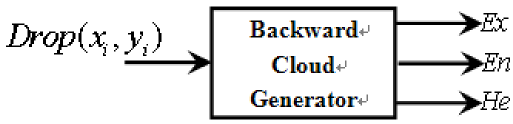

Fig.3 The model of Backward Cloud Generator. By inputting a series of droplets, the cloud model which is described by exception *Ex* and entropy *n*, hyper-entropy *He* is generated.

In fact, the generation of backward normal cloud is based on statistical methods. It generates the numerical characteristics of a concrete concept through calculating the numerical figures of the group of cloud droplets which has been given. The realization of backward normal cloud generator [2] can be described as follows:

Algorithm 2: *BackCloud*(*X*)

Input: a series of M-dimensional cloud droplets,

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{in}), i = 1, 2, \ldots, N$$

Output: The numerical characteristics of the concrete concept, *Ex, En, He.*

Step: Execute the following steps:

step1: compute the mean of samples, $\bar{X} = \frac{1}{n}\sum_{i=1}^{N} x_i$, first-order sample center distance $\frac{1}{N}\sum_{i=1}^{N} |\bar{X} - x_i|$, and sample variance $S^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\bar{X} - x_i)^2$;

step2: Let $Ex = \bar{X}$;

step3: Let $En = \sqrt{\frac{\pi}{2}} \times \frac{1}{N}\sum_{i=1}^{N} |x_i - E_x|$;

step4: Let $He = \sqrt{|S^2 - E_n^{\ 2}|}$.

As seen from algorithm2, the group of cloud droplets is the qualitative description of a concrete concept. After the backward normal cloud generator, the numeric features of qualitative concept are generated from step 2 to step 4. The qualitative description of the concrete concept is the statistical features of the overall group of cloud droplets at a certain extent.

The normal cloud model is widely used in daily life as a result of the normal distribution is the most commonly distribution. Then in the next subsection, the forward and backward normal cloud generator is used simulate the target of several shooters.

C. Experiments with Forward/Backward Cloud Generator

The realization of forward normal cloud generator and backward normal cloud generator can be illustrated by the shooting examples. To shoot the target, the centre is established to coordinate the origin of the Cartesian coordinates. Let a shooter fire 20 times, and record the location data of target landed on the target set of coordinates, then employ forward and backward cloud generator to simulate the shooting 20 and 200 times of the shooter. It is assumptive that there are two shooting cases are as follows:

Shooter A: The distribution of shooting results is showed in Fig. 4A (a). As shown in Fig.4A (a), the distribution is more stable. After we have been computed, its numerical feature is *Ex* = (1.73,-0.75); *En* = (2.91, 1.93); *He* = (0.50, 0.63). The target sets lie in the upper and left of the coordinate.

Shooter B: The distribution of shooting results is shown in Fig.4B (a). We can see from Fig.4B (a), its distribution is more discrete and unstable. The numerical feature is *Ex* = (-1.16, 0.17); *En* = (4.34, 3.39); *He* = (1.90, 1.02).

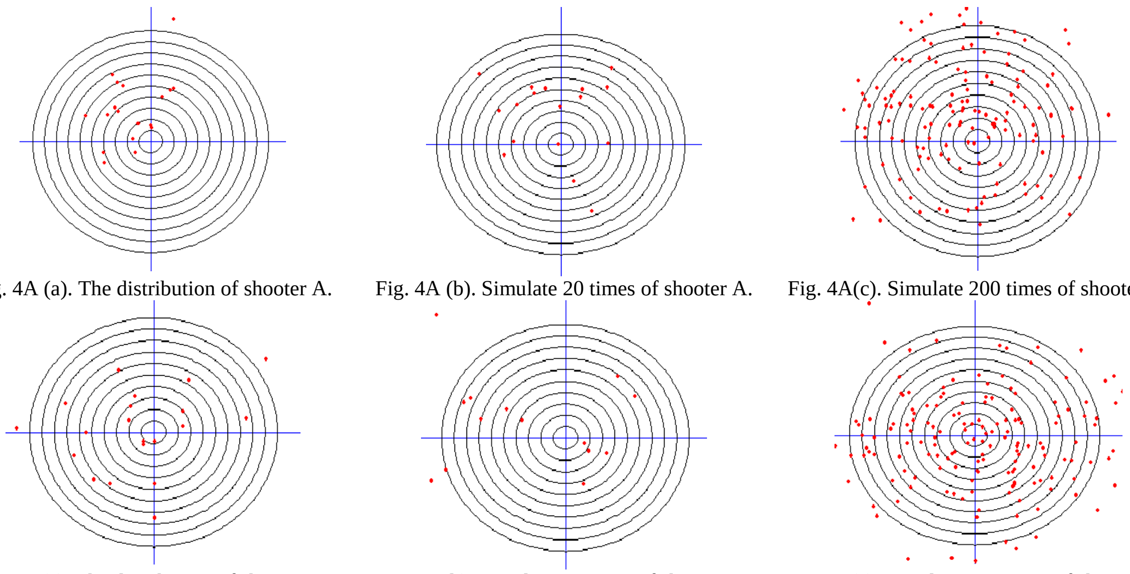

Fig. 4A (a). The distribution of shooter A. Fig. 4A (b). Simulate 20 times of shooter A. Fig. 4A(c). Simulate 200 times of shooter A.

Fig.4B (a). The distribution of shooter B. Fig.4B (b). Simulate 20 times of shooter B. Fig.4B(c). Simulate 200 times of shooter B.

Fig.4. Using forward and backward cloud generator to simulate the target of two shooters.

Then the shooter’s targets are simulated 20 and 200 times shown in Fig.4A (b), Fig.4A (c), Fig.4B (b) and Fig.4B (c).

As seen from target sets of shooter A, the center of target is (-1.5, 3.0). It indicates that the distribution of target set lies at the second quadrant (that is, on the upper and left side of the coordinate). Through the forward normal cloud generator, the entropy is (2.1, 2.9) and the hyper-entropy is (0.36, 1.0). These numerical features indicate that the distribution of target is relatively concentrated, and the y-axis (that is, in the vertical direction) has the relatively larger divergence. From Fig.4A (b) and Fig.4A(c) can be seen, simulated target is in line with the original distribution fired by shooter A in principle. However, from Fig.4A (b) and Fig.4A(c), in the course of a greater reduction target fell outside the target set, this phenomenon in theory and in practice are unreasonable, and the clouds through the use of region partition way to better to resolve this problem. Following are the model-partition two-dimensional thinking, and achieving practical application.

IV. Forward and Backward N-D Cloud Model Based on Region Partition

By making use of the mathematical thinking of region partition, we can use the partial cloud droplets distribution to fit the overall distribution of clouds. For two-dimensional model of normal distribution of clouds, we can divide the regional distribution of clouds into a number of regional distributions, and then use forward or backward cloud generator on each small region to generate the overall distributions.

It may consider a series of discrete region partition through the clouds to find a unified approach to the overall situation of clouds to simulate the real situation. In mathematics [15], it is first through a series of discrete distribution of observations and analysis, based on experience with a broad distribution, selected one of the distributions of partition for this function, to establish the mathematical model, and then calculated the determining factor in the final form of the region partition function proposed. The differences between the known data and partition the function (curve) are as far as possible from a minimum through the eventual adoption of least-squares and other mathematical methods.

A. Backward cloud generator based on region partition

The backward cloud generator divides the concept of a quantitative description into a certain number of sub-spaces and then computes numerical features of each sub-space to get a quantitative description of the concept. The description of concept is a multi-dimension array after dividing, and each one is correspond to the qualitative description of a sub-space dimension.

In the course of its realization, *Ex*= {*Ex(i)* | *i=1,2,…, parts*}, *En*={*En(i)*|*i=1,2,…,parts*} and *He*= {*He(i)*|*i=1, 2,…,parts*} were *parts*-dimensions expectations, entropy and hyper-entropy. *parts* is the number of dimensions of the concept after dividing. The clouds for importation are described as $X=\{X_i \mid x_i=(x_{i_1},x_{i_2},...,x_{i_m}), i=1,2,...,N\}$. The concrete algorithm for realizing can be described as follows:

Algorithm 3: *BackCloudRP* (*X, parts*)
Input: *N*-*m*-dimension Cloud droplets, *parts*
Output: *Ex*, *En*, *He*
Step: Execute the follow steps
step1: if *parts* > *1* then turn step2,
else call *BackCloud*(*X*);
step2: Partition, Divide clouds *X* into *parts* sub-spaces, as $X=\bigcup_{i=1}^{parts} X_i$;
step3: Repeat Step4—Step6 While *i* = *parts*;
step4: For each sub-space, call *BackCloud* (X_i)
step5: *Ex(i) = Ex, En(i) = En, He(i) = He;*
step6: *i = i + 1*;

As mentioned in Algorithm 3, if *parts =1* then there is not any division operation, then call Algorithm 1 to calculate the numerical feature of concept. In step2, it divides the description of concept into *parts* sub-spaces. From step3 to step6, for each sub-space, use Algorithm 1 to compute the numerical feature and insert them into numerical of overall clouds description.

B. Forward cloud generator based on region partition

Being similar to forward normal cloud generator, the cloud droplets are generated for each sub-space, and then combine all the cloud droplets generated to make the overall clouds, the description of concept. *Ex*, *En* and *He* are *parts*-dimension expectation, entropy and hyper-entropy. *N* denotes the number of cloud droplets which will be generated. $Drop(x_i, conf(x_i))$ denotes a cloud droplet where x_i means a m-dimension cloud droplet and $conf(x_i)$ means the confirmation of it. Its realization can be described as follows:

Algorithm 4: *FroCloudRP* (*Ex, En, He, N, parts*)
Input: *Ex*, *En*, *He*, *N*, *parts*
Output: cloud droplets $Drop(x, conf(x))$
Step: Execute the follow steps:
step1: if *parts* > *1* then turn step2
else call *FroCloud* (*Ex*, *En*, *He*, *N*);
step2: Repeat Step3—Step5 While *i* = *parts*;
step3: Call *FroCloud* (*Ex(i)*, *En(i)*, *He(i)*,*N/parts*);
step4: $Drop(x, conf(x)) = Drop(x, conf(x)) \cup Drop(x, conf(x))_i$

where $Drop(x, conf(x))_i$ denotes the cloud droplets generated for *i*th sub-space.
step5: *i = i + 1*;

In Algorithm 4, each sub-space is thought as a separate domain, and then uses Algorithm 1 to generate a certain number of cloud droplets showed from step3 to step5. The overall cloud will be generated by stacking all the cloud droplets together.

C. Description of algorithms

As mentioned in section III, algorithm *FroCloudRP* transfers the qualitative description to quantitative description for the concrete concept, and algorithm *BackCloudRP* turns the quantitative description into qualitative description. These two algorithms based on region partition are multi-dimensional for the qualitative description.

In the implementation of Algorithm 3 and Algorithm 4, it uses algorithm *BackCloud* and *FroCloud* mentioned in the third section. It first divides the domain of concept into several parts, and then uses *BackCloud* to generate the numerical features in each sub-space and *FroCloud* to generate a series of cloud droplets. The number of numerical features for the overall cloud is pre-defined and it is equal to the sub-space number.

There are many methods for dividing the concept domain. The first one views the whole area as the overall

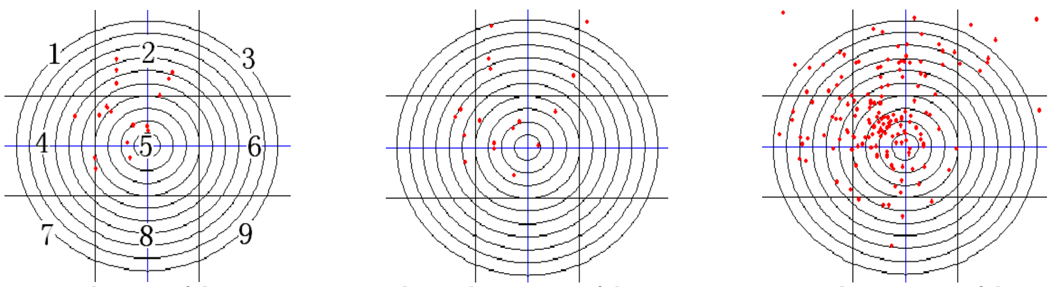

Fig.5a. The target of shooter A. Fig.5b. Simulate 20 times of shooter A. Fig.5c. Simulate 200 times of shooter A.

Fig.5. Divide the target of shooter A into 9 small parts showed in Fig.5a. It view the whole area as the overall space. After computing the numerical feature of each sub-parts, then use forward cloud generator to generate the cloud droplets belonging to each sub-part to simulate 20 and 200 times for overall.

space to divide, while the other method views the area which cloud droplets appear only in as the overall space to divide. And at the same time, the number of sub-space after divided is also an important factor to affect the partition result. In section 5, we use the two division methods to simulated the result for shooter A's target while dividing the area for 9 pieces.

V. EXPERIMENTATION

We use algorithms proposed in section IV to re-simulate the shooter's target. The target is first divided the whole area into 9 small areas (showed in Fig. 5a). Then use Algorithm 3 and Algorithm 4 to simulate shooter A for 20 and 200 times. The simulation result is shown in Fig. 5.

After divided, we used the region partition algorithms proposed in section 4 to simulated the target of shooter A. The numerical feature of each sub-part is shown in Table 1.

TABLE I.
NUMERICAL FEATURES OF EACH SUB-PARTS

Block ID	*Ex*	*En*	*He*	*points*
2	(0.21,6.33)	(2.70,1.88)	(1.24,1.23)	7
4	(-4.80,0.70)	(0.88, 2.00)	(0.46,1.05)	2
5	(-1.89, 1.33)	(1.56,1.55)	(0.52,0.54)	11
1,3,6~9	(0,0)	(0,0)	(0,0)	0

As seen from Table 1, the more points lies in the 2nd, 4th and 5th sub-part, the numerical feature is computed by the backward cloud generator.

Then the number of features shown in the Table (Tab. 1) is used to generate droplets in each sub-space. According to Fig. 5a, the number of points in each small space generated is in proportion to the number of points it has got. For example, to generate a total of 200 cloud droplets of clouds, it should generate 200 * 11 / 20 = 110 cloud droplets of clouds with *Ex* = (-1.89,1.33), *En* = (1.56,1.55) and *He* = (0.52,0.54) in the 5th sub-space. Restore 20 and 200 of the cloud droplet clouds results is shown in Figure Fig. 5b and Fig. 5c.

At the same time, we set the area which has points appeared as the whole space to divide, then it was also divided into 9 pieces. And the simulated result is showed in Fig.6.

As showed in Fig,5 and Fig.6, The overall space is divided into nine sub-spaces numbered from 1 to 9. Then we use algorithms proposed in section 4 to simulated the target of shooter A. The result after simulated is serious different. From Fig.6, the result after simulated is so convergences that there are so many points are coincided together.

A comparison between Fig. 5, Fig. 6 and Fig. 4A indicates that after using the region partition process is better to describe the actual shooter shooting situation. However, there are individual factors affect the accuracy of the simulation results. The analysis of the factors affected the region partition is made in the coming text.

VI. THE ANALYSIS FOR ALGORITHM

As seen from the experiments made above, there are many factors affected the execution of region partition algorithms.

The number of sub-space is the main factor for affecting the implementation of Algorithm 3 and Algorithm 4. Specially, this number has more serious impact for Algorithm 3. In the implementation process of Algorithm 3, it first divides the domain of concept into several sub-spaces, and in the computation of each sub-space, Algorithm 2 is called to calculate numerical feature. Let *T (BackCloud)* denote time consumption while performing Algorithm 2. The time required for implementing Algorithm 3 is *T(BackCloudRP) = parts * BackCloud)).* While the factor for affecting the implementation of Algorithm 4 is the number of cloud droplets wanted to generate. Let *T(FroCloud)* denote time required for performing Algorithm 1. The time required for implementing Algorithm 3 will be *T(FroCloudRP) = T(FroCloud).*

A. Factors affected partition effect

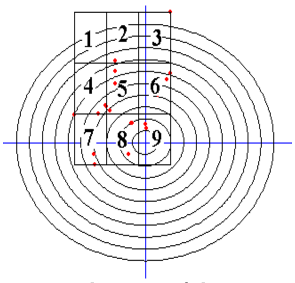

Fig.6a. The target of shooter A.

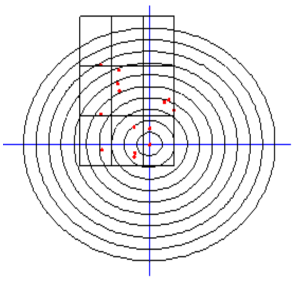
Fig.6b. Simulate 20 times of shooter A.

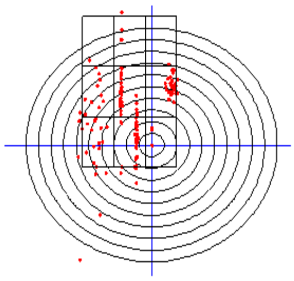
Fig.6c. Simulate 200 times of shooter A.

Fig.6. Divide the target of shooter A into 9 small parts showed in Fig.6a. It view the area where the targets appears as the whole space. After computing the numerical feature of each sub-parts, then use forward cloud generator to generate the cloud droplets belonging to each sub-part to simulate 20 and 200 times for overall.

In the realization of algorithm *FroCloudRP*, the reduction may be the results of simulation of the impact of major factors are:

- *The number of blocks.* In clouds generator, if this number is equals to 1, the algorithm will be degraded in the first two algorithms, Algorithm 1 and Algorithm 2. The experiments show that the number recommended by the strategy to small for the number of square feet for a few in Algorithm 3 and Algorithm 4, and then to carry out region partition. Because of the number of square feet of space a few times, each sub-space is a square space, compared rules, to calculate.

- *The differences of approach of division*. There are at least two different methods to divide the overall space. The first method view whole target as a set of overall space to divide while the other view the distribution of samples as an overall space for division. In the same number of cases, the division granularity of the former is larger than the latter.

It takes shooter A's target into account. The space should be divided into 4, 9 and 16 sub-spaces, and then use region partition algorithms to generate cloud droplets showed in Fig.7. As showed in Fig. 7.1, divide the whole target into several sub-parts while Fig. 7.2 experiment used by the strategy is to cloud droplet mission of the region as a whole in space.

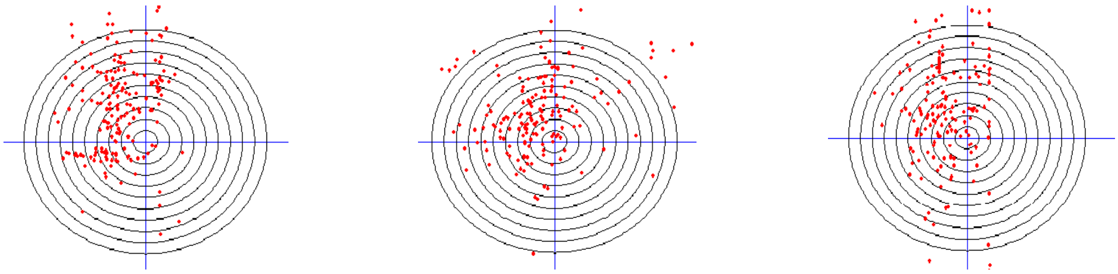

Fig.7.1(a). Divide 4 sub-parts and Simulate 200 times of shooter A. Fig.7.1(b). Divide 9 sub-parts and Simulate 200 times of shooter A. Fig.7.1(c). Divide 16 sub-parts and Simulate 200 times of shooter A.

Fig.7.1. View the whole target set as overall space to divide. It divides 4,9 and 16 sub-parts and then simulates 200 times of shooter A.

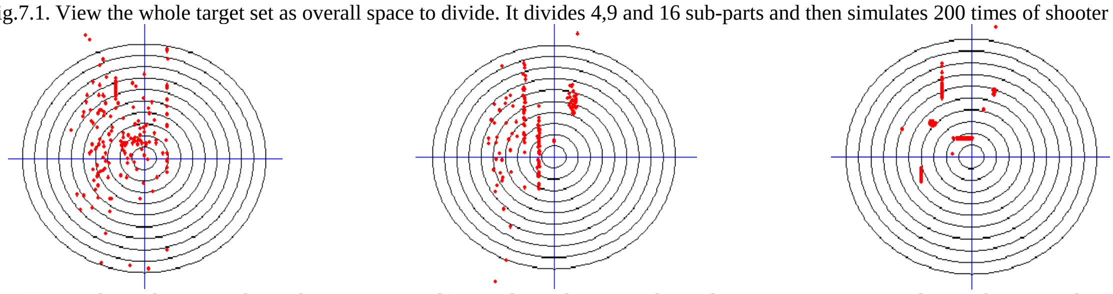

Fig.7.2(a). Divide 4 sub-parts and Simulate 200 times of shooter A. Fig.7.2(b). Divide 9 sub-parts and Simulate 200 times of shooter A. Fig.7.2(c). Divide 16 sub-parts and Simulate 200 times of shooter A.

Fig.7.2. View the whole distribution region of data as overall space to divide. It divides 4,9 and 16 sub-parts and then simulates 200 times of shooter A.

Fig. 7. Simulated 200 times of shooter A after several difference condition of division.

These results show that with the choice of different strategies for space division, the reduction will influence the outcome. It is not satisfactory in Fig. 6.2(c) because the distribution of cloud droplets is too concentrated in local sub-space under this kind of division. The situation is more extreme if the breakdown of each sub-space contains only a point, at *En* = ***0***, and *He* = ***0***, the use of reverse-partition, then there will be *N / parts* coincided with a cloud droplet and a total of parts points.

According to the experimental analysis, this method in practical applications of space taken by the division of general guiding strategy is to convert the whole distribution subject to normal form, also need to avoid excessive local concentration.

B. Evaluation for algorithms

The clouds discussed in this article are subject to the normal distribution, this is because the normal distribution is most common in the daily life of the distribution [3]. In accordance with the laws of normal distribution can be on cloud droplet group qualitative contribution to the concept of volume is defined as follows:

Definition 5: contribution of Cloud [2]. There is an element Δx in small interval on basic variable X . The amount of contribution Δc of this variable for the qualitative concept $\tilde{A}$ is defined as follows:

$$\Delta c \approx \mu_{\tilde{A}}(x)\times\Delta x/\sqrt{2\pi}En$$

Then the amount of contribution for $\tilde{A}$ generated by

TABLE II.
THE PROPORTION OF CLOUD DROPLETS GENERATED BY FORWARD CLOUD GENERATORS

	1	2	3	4	5	6	7	8
No Partitions	55.5	55.5	58.5	52.0	56.5	57.5	55.5	53.0
Partitions (4)	82.5	83.5	83.5	87.5	85.0	89.5	86.5	85.0
Partitions (9)	81.5	79.5	75.0	85.5	82.0	84.0	79.5	79.0
Partitions (16)	90.0	89.5	86.5	92.0	93.0	89.5	91.5	90.5

As seen from Table 2, there are more cloud droplets lies in the interval between *Ex – 3En* and *Ex + 3En*.

the all elements of domain can be defined as follows:

$$C=\frac{\int_{-\infty}^{+\infty}\mu_{\tilde{A}}(x)dx}{\sqrt{2\pi}En}=\frac{\int_{-\infty}^{+\infty}\exp(-(x-Ex)^2/2En^2)dx}{\sqrt{2\pi}En}=1$$

According to Definition 4, it could get $\frac{1}{\sqrt{2\pi}En}\int_{Ex-3En}^{Ex+3En}\mu_{\tilde{A}}(x)dx=99.74\%$ ·

We can see that the distribution of cloud droplets generated in the interval between $Ex-3En$ and $Ex+3En$ is on the great contribution. Take advantage of this characteristic, this paper presents the following evaluation rules:

Rule 1: The vast majority of cloud droplets generated by forward cloud generator should be in the interval between $Ex-3En$ and $Ex+3En$.

It is viewed that numerical feature calculated through backward cloud generator as standard numerical feature. A comparison between Algorithm 1 and Algorithm 4 is proposed in this paper. We use the shooter A's target data provided in the section III to conduct random test eight times independently. It generates 200 cloud droplets each time and calculates the proportion (%) of droplets lied in the region of between $Ex-3En$ and $Ex+3En$. The result is showed in Table 2.

From the experimental results, we can see that the cloud droplets generated by region partition algorithm in a "reasonable" region is significantly higher than the non-region partition algorithm.

VII. CONCLUSION

The Cloud model is a good tool for the conversion between the qualitative and quantitative description of a concrete concept. The proposition of region partition algorithms is the theory of the expanding of cloud theory. The theory has a theoretical value and practical significance. It has a better result in the real world of uncertainty to simulate through the clouds theory.

How to select parameters to avoid the distribution points which are too concentrated in the local area of target will be the key elements we are working on in the future.

ACKNOWLEDGMENT

This work was supported by National Science Foundation Project (*No:60872057*), Natural Science Foundation of Fujian Province of china (*No:A071003*).

REFERENCES

[1] LI De-Yi, LIU Chang-Yu, DU Yi, HAN Xu, "Artificial Intelligence with Uncertainty", *Journal of software*, Vol. 15, pp.1583–1594, Nov. 2004

[2] LI De-Yi, DU Yi, *Artificial Intelligence with Uncertainty*, Beijing: National Defense Industry Press, 2005

[3] LIU Ying-Ming, *Fuzziness, other side of accuracy*, Beijing: Tsinghua University Press, 2001

[4] LI De-Yi, MENG Hai-Jun, SHI Xue-Mei, Membership Clouds and Membership Cloud Generators, *Computer Research & Development*, Vol. 32, pp.15—20, Jun. 1995.

[5] LI De-Yi, LIU Chang-Yu, "Study on the University of the Normal Cloud Model", *Engineering Science*, Vol.6, pp.28—34, Aug.2004

[6] ZHANG Yong, ZHAO Dong-Ning, LI De-Yi, "The Similar Cloud and Measurement Method", Information and Control, Vol.33, pp.129—132, Apr.2004

[7] DAI Chao-Hua, ZHU Yun-Fang, CHEN Wei-Rong, LIN Jian-Hui, "Cloud Model Based Genetic Algorithm and Its Application", *ACTA ELECTRONICA SINIA*, Vol.35, pp.1419—1424, Jul.2007

[8] ZHANG Yong, NIU Xia-Mu, ZHAO Dong-Ning, "A Method of Protecting Relational Database Copyright with Cloud Watermark". *International Journal of Information Technology*, Vol.1, No.3, pp.112—116, 2004

[9] LI De-Ren, DI Kai-Chang, LI De-Yi, "Knowledge Representation and Uncertainty Reasoning in GIS Based on Cloud Model" *in 2000 Proc. 9th International Symposium on Spatial Data Handing*, Beijing, pp.3a.3—14

[10] ZHANG Ke, YUAN Jin-Han, YANG Xue-Ming, "Edge Detection of Images Based on Cloud Model Cellular Automata", *WSEAS Trans. on Biology and Biomedicine*, Vol.5, pp.67—72, May.2007

[11] DI Kai-Chang, LI De-Yi, LI De-Ren, "Cloud Theory and Its Applications in Spatial Data Mining and Knowledge Discovery", Journal of Image and Graphics(A), Vol.4, pp.930—935, Nov.1999

[12] LI Hong-Xing, WANG Pei-ZHuang, Fuzzy Mathematics, Beijing: National Defense Industry Press, 1994

[13] CE Shannon, A Mathematical Theory of Communication, *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol.5, Issue 1, pp.3-55, 2001

[14] QIU Guan-Hua, *Management Decision & Applied Entropy*, Beijing: China Machine Press, 2002

[15] SHI Miao-Gen, GU Li-Zhen, *Foundation of Science and Engineering Computation*, Beijing: Tsinghua University Press, 1999

GOU Jin was born in China in 1978. He received the Ph.D. degree in computer science and technology from Zhejiang University, Hangzhou City, China, in 2006.

He is currently a teacher at the Department of Computer Science, Huaqiao University, China.

His research interests include knowledge fusion, computation intelligence and knowledge engineering.

ZHAO Zhi-Qiang was born in China in 1985, He received the B.Sc. degree from College of Computer Science & Technology of Huaqiao University. He is currently working towards the M.Sc. in Huaqiao University. His research interests include Computation Intelligence and Knowledge Engineering.

CHEN Wei-Bin received the B.Sc. degree in Computer Science from Wuhan University of geodesy and geometrics. He is currently a professor at the College of Computer Science and Technology, Huaqiao University, China.His research interests include Database, Data Mining, Intelligent Web Information Processing.

LUO Wei was born in Mudanjiang, P.R.China in 1976. He has got the M.S. degree in computer sciences, 2002, University of Wisconsin-Madison, located in Madison, WI, United States His previous research fields include molecular enzymology, NMR, bioinformatics, and database; his current research focuses on artificial intelligence and bioinformatics.