2011 10th International Symposium on Distributed Computing and Applications to Business, Engineering and Science

The Research of Q Learning-Based Estimation of Distribution Algorithm

Hu yugang
Information Department
Changzhou Textile Garment Institute
Changzhou china
Huyugang80@163.com

Abstract: **This paper focuses on the theory of estimation of distribution algorithms. First, elaborated the idea of estimation of distribution algorithms, And then for the limitations of solving complex optimization problems,proposed Q Learning-Based Estimation of Distribution Algorithm. The Q learning algorithm is introduced into evolutionary computation, through the Agent and group interaction, to achieve a probability model of adaptive updates. Test functions using six classical comparative experiment, the results show that the algorithm performance is stable, running time is short, with a strong global search ability, is an efficient solving algorithm for function optimization problems.**

Keywords-* *Estimation of Distribution Algorithm, Q Learning, Evolutionary Search

I THE PROPOSED ESTIMATION OF DISTRIBUTION ALGORITHM

In order to overcome the genetic algorithm because of chromosomal rearrangements lead to the chain problem, people ask if you can not use the crossover and mutation operations, but by the optimal solution set from the extracted information, and then use this information to generate new solutions of the probability distribution, which Chain Linkage Leaning. The probability model using constructive thinking into the evolution of computing the theoretical basis for estimation of distribution algorithms.

The concept of estimation of distribution algorithm first proposed in 1996, and after the year 2000 has been developing rapidly. It will build into the probability model and the sampling process of evolution to replace the traditional crossover and mutation. As is the use of probabilistic models to guide the search process, to avoid the blindness caused by chromosomal recombination and random, Thus effectively improving the efficiency of search, Fast, reliable solution to many of the traditional genetic algorithm optimization problem difficult to solve.

II Q LEARNING-BASED ESTIMATION OF DISTRIBUTION ALGORITHM

A Questions

Not difficult to find by analyzing the existing distribution of variables unrelated to the reason why estimation algorithm will often display a poor performance is due to update its probability vector normal to a single fixed strategy, not only can not guarantee that the whole evolution process of the strategy is always effective, Does not take into account the evolution of the gene locus that appears when the difference. If the probability of each gene locus corresponding values can be adaptive in the evolutionary process of updating, it will help to improve the performance of evolutionary search

In order to achieve the adaptive update the probability vector can be associated with a different gene locus Agent, and the selection probability update rules as its action. Thus, the probability value of each update to convert into Agent performs an action. If the group as a further evolution of the environment, each Agent can use reinforcement learning method and the environment interact to find the optimal movement strategy.

Q learning as a typical reinforcement learning algorithm does not need to estimate the environmental model, but an iterative calculation by optimizing the Q function to obtain the optimal movement strategy, which can be selected as the Agent of learning. It is based on this idea, the following proposed Q Learning-Based Estimation of Distribution Algorithm(QEDA).

B Algorithm design

Q Learning-Based Estimation of Distribution Algorithm with the binary coding, population size is N, code length m, the first generation of the probability vector t is denoted by p (t) = (p1 (t), p2 (t), ... pm (t)), Where pi (t) for the first i loci the probability of taking one. The basic flow algorithm consistent with Figure 1, each iteration including selection, construction (updated) probability models and sampling and other operations.

Sampling operation and PBIL, UMDA, etc. the same algorithm, using Monte Carlo method, in accordance with the probability vector N individuals randomly generated. Options selected in addition to the fitness of each generation according to the optimal sub-groups, but also select the worst sub-groups, based selectivity are r (0 <r <1), the size of the sub-groups are $M=\lfloor rN \rfloor$. Updated statistical probability model, respectively, the optimal operation of first and worst in you to take a sub-group the frequency of, denoted by gi (t) and bi (t), then accordingly update the probability pi (t).

Q learning method using the update pi (t), required for each gene locus associated with an Agent, and the evolution of the corresponding groups of bits of each generation as the environment. The definition of state of the environment, should be able to distinguish the gene locus in the evolutionary process in which the different stages. Therefore, according to the gi (t) and bi (t) to define the relationship between the state.

Greater frequency threshold set θ_{high}, θ_{low} smaller frequency threshold, θ_{diff} for the frequency difference

978-0-7695-4415-1/11 $26.00 © 2011 IEEE
DOI 10.1109/DCABES.2011.97

threshold, Agent$_i$ the first t generations of the state are divided as follows.

(l)gi(t)> θ_{high}.and bi(t)> θ_{high} , or gi(t)< θ_{low} and bi(t)< θ_{low};
(2)|gi(t)-bi(t)|> θ_{diff};
(3) Otherwise does not meet the above criteria.

Agent$_i$ the first generation of action t set includes the following probability update rules.

(1) Action 1, the probability decreases
Pi(t+1)=βpi(t) (0)
(2) Action 2, the probability increases
Pi(t+1)=1-β[1-pi(t)] (1)
(3) Action 3, probability values remain unchanged
Pi(t+1)= pi(t) (2)

equation (0) - (2)., i = l, 2, ..., m, β (o <β <l) to adjust the rate. Agenti interact with the environment, you can choose to perform the appropriate action to obtain the next generation of probability pi (t +1). To store Agenti corresponding to a set of Q, the definition of matrix Qi:

$Q_i=[Q_i(s_j,a_k)]_{3\times 3}$ (3)

Algorithm for each iteration, each Agenti □-greedy strategy selection in accordance with action, according to the rewards and status of their conversions update the corresponding Q values. If the first t-1 on behalf of the state when the environment Agenti s, choose action α executed, the first generation of environmental state transition t to s', then press the style update Qi (s, a):

$$Qi\ (s\ ,\ a)\leftarrow Qi\ (s\ ,\ a)+\ \alpha[r_i(t-1)+\gamma \max\ Qi\ (s'\ ,\ a')-\ Qi\ (s\ ,\ a)] \quad (4)$$

$$r_i(t-1)=\begin{cases}1,|\ pi(t)\ -\ gi(t)\ |<|\ pi(t\ -\ 1)-\ gi(t\ -\ 1)| \\ -\ 1, \quad others\end{cases} \quad (5)$$

Where, ri (t-1) for the Agenti in t-1 obtained on behalf of an immediate return.

Q Learning-Based Estimation of Distribution Algorithm is given below the steps. Algorithm to replace the elitist strategy group to ensure the optimal solution search is not degraded.

```
Algorithm1: Q Learning-Based Estimation of Distribution AlgorithmInitialization Qi, i = l, 2, ..., m zero matrix, p (1) == (0.5,0.5, ..., 0.5), t = 1;
While (termination condition is not satisfied algorithm) do
    According to p (t) sampled individuals generate N-1, and t-1 together constitute the best individual on behalf of the current groups. New individual determination of i-bit value is: generate a random number $\xi \in [0,1]$, if $\xi \le$ pi (t) is taken 1, or take 0;
Calculation of N individuals of fitness function and sorting;
M-choose the best and the worst individuals, the frequency of statistics you get a value of gi (t) and bi (t), i = 1,2, ..., m;
For (i-loci associated Agenti, $1 \le i \le m$)
    Recorded before found their t-1 action on behalf of the state s and a, by gi (t) and bi (t) determine the current state s';
By equation (5) calculated an immediate return, according to equation (3.14) update Qi (s, a);
Generate random numbers $\xi i \in [0,1]$, if $\xi i \le$ □, Randomly selected with equal probability of action a ', otherwise select
a '= argmaxQi (s', a');
According to equation (0) - (2) and the action a 'corresponds to the formula, Calculate the new probability value pi (t +1);
End
t←t+1;
End
```

C Improvement Strategies

The algorithm each update probability pi (t) time, Agenti □-greedy strategy to choose an action, that is, the greater the probability of 1-□ select the maximum Q value of the current state of the corresponding action, but with a smaller probability of randomly selected action □ . As □-greedy strategy is not always accept the best action, but increased the probability of random selection, thus contributing to Agent explore new knowledge, than the greedy strategy with better results.

However, the value of using a fixed □ has some limitations. Especially in the Ageni some time after learning, the current strategy is near optimal, if the probability of 1-□ still randomly choose an action, it will have an impact on the convergence of the algorithm. If you can gradually reduce the evolution □ values, will further enhance the Q study the performance of estimation of distribution algorithms. The use of simulated annealing (SA) algorithm MetroPolis criteria to be able to do this, it is the way by reducing the temperature to gradually reduce the probability of receiving inferior solution.

Here are guidelines for improved use of MetroPolis Q Learning-Based Estimation of Distribution Algorithm

```
Algorithm2: Improved Q Learning-Based Estimation of Distribution Algorithm
Initialization Qi,, i = 1,2, ..., m zero matrix, the temperatureτ=τ0, p (1) = (0.5,0.5, ... 0.5), t = 1;
While (termination condition is not satisfied algorithm) do
According to p (t) sampled individuals generate N-1, and t-1 together constitute the best individual on behalf of the current population;
Calculation of N individuals of fitness function and sorting;
M-choose the best and the worst individuals, the frequency of statistics you get a value of gi (t) and bi (t), i = 1,2, ..., m;
For (i-loci associated Agenti, $1 \le i \le m$)
Recorded before found their t-1 action on behalf of the state s and a, by gi (t) and bi (t) determine the current state s';
By (5) calculation of an immediate return, according to equation (4) Update Qi (s, a);
Come to a '= argmaxQi (s', a''), randomly select an action ar, according to the following probability to determine a ':
```

$P\{a'=a^{r}\}=e^{\frac{Q(s,a^{r})-Q(s,a^{*})}{\tau}}$ (6)

$P\{a'=a^{*}\}=1-P\{a'=a^{r}\}$ (7)

According to equation (0) - (2) and the action a 'new formula to calculate the corresponding probability value pi (t +1);

End

Cool: $\tau \leftarrow \lambda\tau$;

$t \leftarrow t + 1$

End

Algorithm to geometric cooling strategy $\tau \leftarrow \lambda\tau$, Where $\lambda \in (0,1)$ is the temperature coefficient。As the temperature decreases , Agenti randomly selected probability of action will become increasingly smaller, When the temperature tends to 0, the strategy is equivalent to the greedy strategy.

III COMPARATIVE EXPERIMENT

A Test Functions

To evaluate the performance of Q Learning-Based Estimation of Distribution Algorithm, The following algorithm using the UMDA, PBIL algorithm, MIMIC algorithm And genetic algorithm function optimization comparative experiments. Select 6 Benchmark test functions for testing. Are the Sphere function, Quadric function, Schaffer function, Griewank function, Rosenbrock function and Rastrigin function. Benchmark functions of these different patterns, with good test performance. Which, Schaffer function, Griewank function and Rastrigin functions are multimodal function, there are a lot of local minima, generally more difficult to find the global optimum algorithm, the algorithm used to test the ability to jump out of local optimum; Rosenbrock function as a single Peak, non-convex pathological function of the flat trend in the value range, the convergence to the global optimal point is remote, can be used to evaluate the efficiency of the algorithm; Sphere function and the Quadric function is a single peak function, can test the accuracy of optimization algorithm, Examine the implementation of the algorithm performance.

B Experimental Analysis

After several tests, Q Learning-Based Estimation of Distribution Algorithm parameters are as follows: frequency threshold wereθ$_{high}$ = 0.75, θ$_{low}$ = 0.25, θ$_{diff}$== 0.35, select the rate of γ = 0.2, probability adjusted rate β = 0.9, learning factor a = 0.2, discount factor = 0.9. For comparison UMDA PBIL algorithm selection algorithm and the rate is taken 0.2, PBIL learning rate algorithm to take 0.1; MIMIC algorithm selection ratio is 0.4; genetic algorithm uses single point crossover, crossover rate 0.7, mutation rate 0.1; their Elitist strategy are used. All of the above algorithm population size are set to 50, the termination condition for the search to the global optimum or the maximum evolution generation T, take T = 200.

Taking into account all the above algorithm has a certain randomness, use them to function f1-f6 are independently tested 50 times, the experimental results shown in the table. Among them, Table 1 for each algorithm the number of global optimal value obtained, Table 2 shows the results of 50 runs on average, standard deviation and worst values, Table 3 shows the average running time of each method (unit: Seconds). Table □-QEDA and M-QEDA represent the use of strategies and MetroPolis □-greedy Q Learning-Based Estimation of Distribution Algorithm, the former taking □ = 0.1, initial temperature of the latter: τ0 = 50, temperature coefficient of λ = 0.9.

Table 1 Algorithm for number of times the global optimal value obtained

Function	GA	UMDA	PBIL	MIMIC	□-QEDA	M-QEDA
Sphere	22	27	29	31	42	50
Quadric	23	17	31	24	45	50
Schaffer	11	11	27	12	40	50
Griewank	28	24	34	29	45	50
Rosenbrock	14	31	49	28	46	50
Rastrigin	19	21	35	14	44	50

Table 2 The results of the algorithm is run 50 times the mean, standard deviation and worst values

Function		GA	UMDA	PBIL	MIMIC	□-QEDA	M-QEDA
Sphere	average	1.1200E-03	9.2000E-04	8.4000E-04	4.5600E-03	3.2000E-04	0
	Standard deviation	1.0029E-03	1.0069E-03	9.9714E-04	1.9290E-02	7.4066E-04	0
	Worst value	2.0000E-03	2.0000E-03	2.0000E-03	1.2800E-01	2.0000E-03	0
Quadric	Average	0.5166	1.1365	0.1091	0.3731	0.0287	0
	Standard deviation	2.5801	3.4736	0.1407	1.1744	0.0870	0
	Worst value	18.3680	23.2470	0.2870	7.7150	0.2870	0
Schaffer	average	5.7299E-03	3.1967E-03	5.5335E-04	4.1733E-03	3.8145E-OS	0
	Standard deviation	5.0882E-03	4.1418E-03	1.5937E-03	4.9006E-03	7.7064E-05	0
	Worst value	1.2625E-02	1.4193E-02	9.5638E-03	1.8953E-02	1.9072E-04	0
Griewank	average	0.3550	0.4627	0.2582	0.3478	0.0807	0
	Standard deviation	0.4045	0.4572	0.3802	0.4151	0.2445	0
	Worst value	0.8068	1.1645	0.8068	1.0400	0.8068	0
Rosenbrock	average	5.9197	0.5817	0.0038	1.3108	0.0152	0
	Standard deviation	7.7090	1.4511	0.0266	4.0439	0.0521	0
	Worst value	19.5757	7.7538	0.1881	19.5757	0.1957	0
Rastr	average	9.3978	3.4206	0.1190	2.3572	0.0476	0

igin	Standard deviation	14.5125	6.2133	0.1836	5.6836	0.1302	0
	Worst value	79.9851	20.7851	0.3967	20.0000	0.3967	0

Table 3 The average running time of each algorithm

function	GA	UMDA	PBIL	MIMIC	⍰-QEDA	M-QEDA
Sphere	1.0913	1.4806	1.3759	7.5172	1.3169	0.6519
Quadric	1.1619	2.1466	1.2916	10.0963	1.1741	0.6056
Schaffer	1.0669	0.3894	0.2441	0.9788	0.2641	0.1656
Griewank	0.8034	0.8894	0.5781	3.1812	0.6056	0.3887
Rosenbrock	1.2272	1.2347	0.1844	8.6994	1.1469	0.5784
Rastrigin	1.1497	1.9047	1.0859	13.7091	1.2369	0.9159

It can be seen, both the traditional genetic algorithm, or UMDA, PBIL and MIMIC other existing distribution algorithm, function optimization in solving these complex problems are not easy to search the global optimum value. Which, PBIL search success rate of slightly higher average of 68%, other 3, the algorithm less than 50%. Q Learning-Based Estimation of Distribution Algorithm are demonstrated excellent performance, especially after using Metropolis criterion and, for the 6 functions are 100% Benchmark global optimal value obtained. In the algorithm execution time, the dual variable associated MIMIC worst performance of the algorithm, generally longer than the other algorithms 5-10 times; and M-QEDA algorithm in addition to solving the Rosenbrock function as PBIL algorithm, but in other cases have shown The best time performance.

References

[1] Sergios T, Konstantinos K. Pattern Recognition, Second Edition [M]. San Diego: Academic Press. 2003.

[2] Dasgupta D, ,Forrest S. Artificial immune systems and their applications [M]. Berlin: Spring-Verlag 1998

[3] Slowinski R, Hapke M. Scheduling under fuzziness [M]. New York: Physica-Verlag, 2000.

[4] Seber G A R Linear regression analysis [M]. New York: John Wiley, 1977.

[5] Deng J L. Introduction to grey system theory (J]. The Journal of Grey System, 1989, 1(1):