Pattern Recognition, Vol. 24, No. 10, pp. 1003–1008, 1991
Printed in Great Britain

0031-3203/91 $3.00 + .00
Pergamon Press plc

A SIMULATED ANNEALING ALGORITHM FOR THE CLUSTERING PROBLEM

SHOKRI Z. SELIM and K. ALSULTAN

Department of Systems Engineering, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia 31261

(*Received* 19 *September* 1989; *in revised form* 28 *September* 1990; *received for publication* 20 *March* 1991)

Abstract—In this paper we discuss the solution of the clustering problem usually solved by the K-means algorithm. The problem is known to have local minimum solutions which are usually what the K-means algorithm obtains. The simulated annealing approach for solving optimization problems is described and is proposed for solving the clustering problem. The parameters of the algorithm are discussed in detail and it is shown that the algorithm converges to a global solution of the clustering problem. We also find optimal parameters values for a specific class of data sets and give recommendations on the choice of parameters for general data sets. Finally, advantages and disadvantages of the approach are presented.

Fuzzy cluster analysis Simulated annealing Global algorithms

1. INTRODUCTION

In this paper we consider the problem of clustering n data points, patterns, into C clusters. In particular we will give an algorithm which attempts to find the optimal solution to the following mathematical program:

minimize

$$J(W, Z) = \sum_{i=1}^{n} \sum_{j=1}^{C} w_{ij} d_{ij}^2,$$

subject to

$$\sum_{j=1}^{C} w_{ij} = 1, 1 \le i \le n$$

$$w_{ij} = 0 \text{ or } 1, 1 \le i \le n, 1 \le j \le C,$$

where d_{ij} denotes the Euclidean distance between pattern i and center of cluster j, and

$$w_{ij} = \begin{cases} 1 & \text{if pattern } i \text{ is assigned to cluster } j \\ 0 & \text{otherwise.} \end{cases}$$

The above optimization problem is a nonconvex program[1] and it has local minimum solutions which may not be global. The well-known K-means algorithm and variants of it are usually used to solve the above problem.[2,3] The solution obtained usually depends on the initial clustering used in the algorithm. The K-means algorithm could yield solutions which are not even a local minimum of the optimization problem. Selim and Ismail[4] showed that the K-means algorithm converges in a finite number of iterations, and gave conditions under which the obtained solution is a local minimum.

In this paper we describe a simulated annealing-based algorithm for solving the above clustering problem. Klein and Dubes[5] reported on experiments in projection and clustering by simulated annealing. They concluded that research on the choice of the simulated annealing parameters, cooling schedules, is needed. In Section 2 we describe the annealing process in thermodynamics and show the analogy with the clustering process. In Section 3 the details of a simulated annealing algorithm for the clustering problem are given. The parameters required by the algorithm are discussed in Section 4, where we also discuss convergence of the algorithm to a global solution of the optimization problem. In Section 5 we recommend a set of parameters for a special type of data set. In Section 6 we give guidelines for selecting parameter values for the case of general data. We end this paper with a discussion of the advantages and disadvantages of the proposed algorithm.

2. THE ANNEALING PROCESS IN THERMODYNAMICS

Annealing refers to the process of heating up a solid to a high temperature followed by slow cooling achieved by decreasing the temperature of the environment in steps. At each step the temperature is maintained constant for a period of time sufficient for the solid to reach thermal equilibrium. At equilibrium the solid could have many configurations, each corresponding to different spins of its electrons and to a specific energy level. At equilibrium the probability of a given configuration, $P_{\text{configuration}}$, is given by Boltzmann distribution:

$$P_{\text{configuration}} = k \exp(-E_{\text{configuration}}/T)$$

Table 1. Analogy between the simulated annealing process and the clustering problem

Simulated annealing	The clustering problem
1. The solid being annealed	The optimization problem
2. The current configuration	The current assignment of patterns to clusters
3. Perturbation of the current configuration	Generating a new assignment
4. The trial configuration	The trial assignment
5. The energy associated with a configuration	The function J associated with an assignment
6. Accepting a new configuration if its energy level is less than the current energy level	Accepting an assignment which results in an improvement in the function J
7. Acceptance of a higher energy configuration with some probability	Acceptance of an assignment with a higher value of J, hoping that a much better assignment becomes reachable

where $E_{\text{configuration}}$ is the energy of the given configuration, T is the temperature and k is a constant.

Metropolis *et al.*[6] proposed a Monte Carlo method to simulate the process of reaching thermal equilibrium at a fixed value of the temperature T. In this method a randomly generated perturbation of the current configuration of the solid is applied so that a trial configuration is obtained. Let E_c and E_t denote the energy level of the current and trial configurations, respectively. If $E_c > E_t$, then a lower enery level has been reached, and the trial configuration is accepted and becomes the current configuration. On the other hand, if $E_t \geq E_c$ then the trial configuration is accepted as the current configuration with probability proportional to $\exp(-(E_t - E_c)/T)$. The process continues where a transition to a configuration of a higher energy level is not necessarily rejected. Eventually thermal equilibrium is achieved after a large number of perturbations, where the probability of a configuration approaches Boltzmann distribution. By gradually decreasing T and repeating Metropolis simulation, new lower energy levels become achievable. As T approaches zero least-energy configurations will have a positive probability of occurring.

Kirkpatrick *et al.*[7] and Černy[8] pointed to the analogy between the simulation of the annealing process and the optimization problem. To be specific, we describe this analogy in the case of the clustering problem on hand (see Table 1).

3. A SIMULATED ANNEALING ALGORITHM FOR THE CLUSTERING PROBLEM

In this section we describe a simulated annealing algorithm for solving the clustering problem of Section 1. In the algorithm the following notation is used:

a_i denotes the cluster to which data point i is assigned, i.e. $w_{i,a_i} = 1$ and $w_{ij} = 0$ for $j \neq a_i$,

$A \in \mathbf{R}^n$ denotes the vector whose components are a_i,

$U(0, 1)$ denotes the uniform distribution with domain $[0, 1]$,

The subscripts c and t denote the current and trial assignments, respectively, while the subscript b denotes the assignment that gives the least value of the function $J(W, Z)$ so far.

3.1. *Statement of the algorithm*

Initialization. Set the values of T_1, ε and μ, to prespecified initial temperature, final temperature and temperature multiplier, respectively. Start with an arbitrary grouping of the data points into C clusters. Let the vectors A_b and A_c denote this assigment. Compute the corresponding criterion function $J(W, Z)$. Set the scalars J_b and J_c to this value. Let $T = T_1$, and go to Step 1.

Step 1. Obtain a trial assignment A_t. Let J_t be the corresponding function value.

Step 2. If $J_t > J_c$ go to Step 3, otherwise accept this trial assignment, let $A_c = A_t$, and $J_c = J_t$. If $J_t \geq J_b$ replace *count* by *count* + 1 and go to Step 4. Otherwise, let $J_b = J_t$, $A_b = A_t$, *count* = 0 and go to Step 4.

Step 3. Draw a random number $y \sim U(0, 1)$. If $y > \exp(-(J_t - J_c)/T)$ go to Step 4 (trial assignment is not accepted); otherwise, let $J_c = J_t$, $A_c = A_t$ and go to Step 4.

Step 4. If *count* $< N$ go to Step 1. Otherwise replace T by μT. If $T < \varepsilon$ stop otherwise go to Step 1.

In Step 1 of the above algorithm a trial assignment of the patterns to the clusters is generated. In the following we discuss the details of this step.

3.2. *Obtaining a trial assignment*

A trial asignment should be in some neighborhood of the current assignment. A neighbor of a given assignment can be obtained in various ways. There is no mathematical definition that could be used as a guideline. However, the neighboring assignment should be in some sense "close" or related to the current assignment.

The following procedure is proposed for generating an assignment which is a neighbor of a current assignment.

An algorithm for obtaining a neighboring assignment.

Step 1. Start with the first pattern, set $i = 1$ and *flag* = *false*.

Step 2. Draw a random number $u \sim U(0, 1)$. If $u > P$ (a prespecified probability threshold (Section 4.5)) set *flag* = *true* and go to Step 4. Otherwise, keep pattern i assigned to its current cluster a_i.

Step 3. If $i = n$ and *flag* = *true* stop. Otherwise, replace i by $i + 1$ and go to Step 2.

Step 4. Let $S_i = \{l | 1, 2, \ldots, C, l \neq a_i\}$. Generate randomly an element l of the set S_i. Assign pattern i to cluster l, i.e. set $a_i = l$. Go to Step 3.

The above algorithm will stop if all patterns have been considered and at least one has been assigned to a new cluster. The closeness between an assignment and its neighbor is controlled by the probability threshold P which is explained in the next section.

4. PARAMETERS OF THE ALGORITHM

The algorithms in the previous section require the following parameters: the initial temperature T_1, the temperature multiplier μ, the probability threshold P, and the number of iterations for reaching equilibrium at a given temperature N. In the following sections we discuss the significance of each of these parameters.

4.1. *The initial temperature of the annealing process*, T_1

If $J_t > J_c$, i.e. the objective function $J(W, Z)$ corresponding to a trial assignment exceeds that of the current assignment then as per Step 3 of Section 3.1 the trial assignment is accepted with probability $\exp(-(J_t - J_c)/T)$. If T is large, the probability of acceptance will be large. Initially, one would like to accept assignments of higher function value to explore a large number of assignments. Hence the initial temperature is chosen as large as possible.

4.2. *The temperature multiplier*, μ

If the best function so far, J_b, does not improve for a certain number of iterations (see Section 4.4), it means that equilibrium has been reached for this temperature. The temperature should be further decreased. This is achieved by multiplying the current temperature by μ, where $0 < \mu < 1$.

4.3. *The probability threshold*, P

This threshold is used in the algorithm for generating a trial assignment (Section 3.2). Note that the probability that assignment A_t differs from A_c in exactly r components is given by:

$$\binom{n}{r} P^{n-x}(1 - P)^r, \tag{1}$$

and hence the average number of changed components is $nP(1 - P)$. Hence for $P > 0.5$, as P increases, the average number of changes decreases. So if one desires to obtain trial assignments which are reasonably close to the current assignment, one would increase P, and vice versa.

The threshold P will be set at two values P_1 and P_2, $0 < P_1 < P_2 < 1$. $P = P_1$ is to be used in the early iterations, so large numbers of changes in the assignments are achieved. After SP iterations, P_2 will replace P_1 in order to make the generated trial assignments closer to the current ones.

4.4. *The number of iterations for reaching equilibrium*, N

If during the last N iterations at a fixed temperature the value of J_b does not improve, it will be assumed that equilibrium has been reached for this temperature level. So in the next iteration, the temperature is reduced by multiplying the current temperature by μ.

We end this section with a discussion on the convergence of the global solution of the optimization problem on hand.

4.5. *Convergence of the simulated annealing algorithm*

Results on the convergence of the simulated annealing process are given in reference (9). First we introduce the notation for this section.

Let A_r refer to a particular assignment of patterns to clusters,

Ω be the set of all possible assignments,

$G_{rt}(T)$ be the probability of generating assignment A_r from assignment A_r at temperature T,

$\rho_{rt}(T)$ be the probability of accepting assignment A_t once it has been generated from A_r at temperature T, and

J_r be the objective function corresponding to assignment A_r.

The following theorem asserts that a simulated annealing algorithm will reach the optimal solution to the underlying problem if it satisfies some conditions.

Theorem 1. A simulated annealing algorithm converges to the solution if the following conditions are satisfied:

(1) the function $G_{rt}(T)$ is independent of T,

(2) $G_{rt} = G_{tr}$ for all A_r, $A_t \in \Omega$,

(3) if $J_r \leq J_t \leq J_u$ then $\rho_{ru}(T) = \rho_{rt}(T)\rho_{tu}(T)$ for all A_r, A_t and $A_u \in \Omega$,

(4) if $J_r \geq J_t$ then $\rho_{rt}(T) = 1$ for all A_r, $A_t \in \Omega$,

Table 2

Case	ρ_{rt}	ρ_{tu}	ρ_{ru}
$J_r < J_t < J_u$	$\exp(-(J_t - J_r)/T)$	$\exp(-(J_u - J_t)/T)$	$\exp(-(J_u - J_r)/T)$
$J_r = J_t < J_u$	1	$\exp(-(J_u - J_t)/T) = \exp(-(J_u - J_r)/T)$	$\exp(-(J_u - J_r)/T)$
$J_e < J_t = J_u$	$\exp(-(J_t - J_r)/T) = \exp(-(J_u - J_r)/T)$	1	$\exp(-(J_u - J_r)/T)$
$J_r = J_t = J_u$	1	1	1

(5) if $J_r < J_t$ then $\rho_{rt}(T) > 0$ for all $A_r, A_t \in \Omega$ and $T > 0$,

(6) If $J_r < J_t$ then $\lim_{T\to 0} \rho_{rt}(T) = 0$ for all A_r, $A_t \in \Omega$.

Proof. See van Laarhoven and Aarts.[9]

In the following theorem it is shown that the simulated annealing algorithm proposed in this paper satisfies all the above conditions needed to converge to an optimal solution of the clustering problem on hand.

Theorem 2. The simulated annealing algorithm of Section 3.1 satisfies the conditions of Theorem 1.

Proof. $G_{rt}(T)$ is the probability that the algorithm in Section 3.2 generates assignment A_t from assignment A_r. To verify that this probability is independent of the temperature T one needs to examine the exogenous parameters of this algorithm. These parameters are C and P. None of them is dependent upon T. Hence Condition (1) is satisfied.

The probability of generating assignment A_t from A_r or vice versa is a function of the number of different pattern assignments as given by (1). Hence $G_{rt} = G_{tr}$ and Condition (2) is satisfied.

To examine Condition (3) one has to enumerate all the possibilities of $J_r \le J_t \le J_u$ as shown in Table 2.

For each case in Table 2 we find the product of entries in columns ρ_{rt} and ρ_{tu} equal the entry in column ρ_{ru}. Hence Condition (3) is satisfied. Condition (4) is satisfied by Step 2 of Section 3.1. Condition (5) is satisfied by Steps 2 and 3 of Section 3.1. Finally, to show Condition (6) note that $\lim_{T\to 0} \rho_{rt}(T) = \lim_{T\to 0} \exp(-(J_t - J_r)/T) = 0$, where the last equality follows if $J_r < J_t$. This completes the proof.

The above conditions do not address the rate of convergence to the global solution, however. In the next section we consider a set of clustering data and give the corresponding set of recommended parameter values.

5. OPTIMAL PARAMETERS

Clusters formed by data have a wide variety of shapes and types. For each type some parameters might give better results than others. By better results we mean that the probability of obtaining the global solution is large enough and/or it is attainable after a reasonable number of iterations. To be able to find the optimal values of the parameters of the algorithm, we considered a class of data which we call Standard Data.

Definition 1. Standard Data satisfies the following conditions:

(1) the clusters are spherical in shape,
(2) the clusters are not overlapping,
(3) the clusters have almost the same number of patterns,
(4) the clusters have almost the same volume.

The above definition does not account for the dimension of the space, i.e. the number of features, nor does it account for the between-centers distances.

We generated Standard Data in 2D space with varying between-centers distances. The solution of each generated problem was thus known. Then we started an expedition searching for the best parameters to use for Standard Data.

5.1. *The search for the best parameters*

Only certain discrete values of the parameters have been selected for the search for the best estimates of parameters. The choice is based on the fact that the algorithm is heuristic and a small change in a parameter may not give a significant difference in the solution. Moreover, convenience to the user has been considered in the selection of the parameters. The candidate values of the parameters used in the search are as follows:

$T_1 \in \{1, \ldots, 9, 10, 50, 100, 500\}$,
$\mu \in \{.3, .4, .5, .6, .7, .8, .9\}$,
$P_1 \in \{0.8, 0.85, 0.90, 0.95\}$,
$P_2 \in \{0.90, 0.95\}$,
$SP \in \{500, 1000\}$ (see Section 4.3), and
$N \in \{10, 20, 30, 40, 50, 100, 200, 300, 400, 500, 1000\}$.

A preliminary search was conducted to obtain the best parameters combination. This search has been performed by solving 10 randomly generated Standard Data sets with the criterion being the number of times the simulated annealing algorithm obtained the exact solution. In the case of ties occurring, they are broken by choosing the parameters that require the smallest average number of iterations to obtain the solution. After obtaining an initial estimate of

Table 3. Optimal parameters for Standard Data

No. of clusters C	No. of patterns n	Initial temperature T_1	Temperature multiplier μ	Iterations to check equilibrium N	Probability threshold P
2	20	3	0.4	20	0.95
	40	3	0.5	50	0.95
	60	3	0.6	100	0.95
	80	3	0.8	200	0.95
	100	3	0.9	200	0.95
3	20	5	0.6	50	0.95
	40	5	0.7	100	0.95
	60	5	0.8	100	0.95
	80	5	0.9	200	0.95
	100	7	0.9	200	0.95
4	20	5	0.7	50	0.95
	40	7	0.9	100	0.95
	60	7	0.9	100	0.95
	80	10	0.9	200	0.95
	100	10	0.9	400	0.95

parameters, a more restricted search was performed by solving 100 problems for each selected combination of parameters.

Optimal parameters for different combinations of number of patterns and clusters are shown in Table 3.

5.2. *Interpretation of the best parameters*

5.2.1. *The initial temperature*, T_1. As indicated in Section 4.1 the temperature controls the acceptance of a trial assignment. The initial temperature is insensitive to the number of patterns for the case of two and three clusters. However, in the case of four clusters the best initial temperature changes with a change in the number of patterns. In general as the clustering problem becomes harder, i.e. as n and C increase, the initial temperature increases, as expected.

5.2.2. *The temperature multiplier*, μ. As the difficulty of the problem increases, the closer μ is to 1, so the temperature steps are closer. As it was expected from the analogy with the annealing process, the temperature multiplier is high. In annealing terminology, this corresponds to slowly cooling the solid which results in a better ground state. This is in contrast to rapid cooling (quenching) corresponding to a small temperature multiplier, which results in a metastable state far from the optimal arrangement of electrons of the solid. Thus the slower the cooling rate, the better the annealing result. However, the annealing process takes a very long time, and correspondingly the clustering problem takes many iterations to reach its optimal solution.

5.2.3. *The probability threshold*, P. As stated in Section 4, this parameter controls the closeness of a trial assignment to a current one. The search started with the idea of having two probability thresholds P_1 and P_2 $(0 < P_1 < P_2 < 1)$, where P_1 is used first to cause a high "shaking" of the assignments at the early stages of the solution. However, the result shows that the values $P_1 = P_2 = P = 0.95$ are the best parameters, and thus there should be only one probability threshold. Hence the optimal value of SP is zero. The high value of P is expected, because the trial assignment should be close to the current one, and this corresponds to what really happens in the annealing process where the configuration of the object changes randomly to a close configuration.

5.2.4. *The number of iterations for reaching equilibrium*, N. At each temperature in the annealing process, the solid is allowed to cool until thermal equilibrium is reached. In the algorithm, assignments are generated until there is no improvement of the objective function for a certain number of iterations (corresponding to a certain time in the annealing process). In this case, the temperature is lowered and the process is repeated. In general, better results are obtained if a large number of iterations is allowed before decreasing the temperature.

6. GUIDELINES FOR SELECTING PARAMETERS FOR GENERAL DATA SETS

A large number of arbitrary data sets was generated. The data violated at least one characteristic of Standard Data. We were able to make the following conclusions and recommendations.

(1) The higher the temperature multiplier μ, the better the solution obtained from the algorithm, but the longer the time it takes the algorithm to solve the problem. Thus the value of μ must be the minimum that can do the job. Generally, $0.7 \leq \mu \leq 0.9$. The choice of this parameter generally depends on

the size of the problem, and the larger the problem the larger the value of μ.

(2) The probability threshold must be high enough to make the neighboring assignment close, but not too close, to its current assignment. We recommend $P = 0.95$.

(3) The higher the number of iterations to detect equilibrium conditions, N, the better the result, but the longer the time it takes the algorithm to solve the problem. N depends on the size of the problem. N in the range [50,600] is recommended.

(4) The initial temperature depends on the magnitude of the objective function of the problem. The higher the magnitude of the objective function, the higher the initial temperature should be. We recommend $T_1 = 10$.

7. CONCLUSIONS

The advantages of the annealing algorithm are as follows.

(1) The algorithm does not "stick" to a local optimal solution, rather it obtains the optimal solution as shown in Theorem 1. An intuitive explanation is that this is because as long as the temperature is higher than zero, ε in the algorithm, there is always a positive probability (though it might be very small for small values of temperature) of getting out of a local minimum. This is in contrast to the K-means algorithm as well as general optimization algorithms, which may stop at a local optimal solution.

(2) There is a relation between the number of iterations and the quality of the solution produced. Thus the user has the ability of choosing the number of iterations to run the algorithm depending on his/her required quality of solution. This property is not enjoyed by the K-means or its variants.

(3) The algorithm could easily be modified to give those assignments which have objective function values close to the optimal solution. This gives the user more flexibility in choosing the solution that satisfies other qualitative aspects not incorporated into the problem.

(4) The algorithm requires very small computer storage.

A disadvantage of the simulated annealing approach is that no characterization of a stopping point is computationally available. Another disadvantage is that verifying that a set of data is Standard Data is as difficult a task as that of solving the clustering problem itself. The user may consider the guidelines of Section 6 if he/she suspects that the data is Standard Data.

REFERENCES

1. S. Z. Selim, Using nonconvex programming techniques in cluster analysis, *Jt Meet. Ops Res. Soc. Am. and Inst. Mgmt Sci.*, Houston, Texas (1981).
2. R. A. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. Wiley, New York (1973).
3. H. Spath, *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*. Wiley, New York (1980).
4. S. Z. Selim and M. A. Ismail, K-Means-type algorithms: generalized convergence theorem and characterization of local optimality, *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 81–87 (1984).
5. R. W. Klein and R. C. Dubes, Experiments in projection and clustering by simulated annealing, *Pattern Recognition* **22**, 213–220 (1989).
6. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller, Equations of state calculations by fast computing machines, *J. chem. Phys.* **21**, 1087–1982 (1953).
7. S. Kirkpatrick, C. D. Gelatt Jr and M. P. Vecchi, Optimization by simulated annealing, *Science* **220**, 671–680 (1983).
8. V. Černy, Thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm, *J. Optimization Theory Applic.* **45**, 41–51 (1985).
9. P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*. D. Reidel Publishing Co., Dordrecht, Holland (1987).

About the Author—SHOKRI Z. SELIM was born in Alexandria, Egypt. He received the B.Sc. degree in Mechanical Engineering and the M.Sc. degree in Industrial Engineering from Cairo University, Egypt, and his Ph.D. degree in Operations Research from Georgia Institute of Technology, Atlanta, in 1970, 1973 and 1979, respectively. Since 1979 he has been a faculty member in the Department of Systems Engineering at King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, where he is at present an Associate Professor. In the Academic year 1988/89 Dr Selim was a Visiting Professor in the Department of Systems Design Engineering at the University of Waterloo.

Dr Selim's research interests are in the areas of cluster analysis, simulation of large systems, location-allocation problems and parameter estimation.

About the Author—K. ALSULTAN was born in Algaseem, Saudi Arabia. He received the B.Sc. and M.S. degrees from King Fahd University of Petroleum and Minerals, Saudi Arabia, and his Ph.D. degree from the University of Michigan, Ann Arbor, in 1985, 1987 and 1990, respectively, in Industrial Engineering. His contributions in this paper are in Sections 5, 6 and 7.