

A Simulated Annealing Algorithm for the Circles Packing Problem

Defu Zhang¹ and Wenqi Huang²

¹ Department of Computer Science, Xiamen University, 361005, China
dfzhang@xmu.edu.cn

² School of Computer Science, Huazhong University of Science and Technology,
Wuhan 430074, China

Abstract. We present a heuristic simulated annealing algorithm to solve the circles packing problem. For constructing a special neighborhood and jumping out of the local minimum trap, some effective heuristic strategies are presented. These strategies are from nature and can allow the iterative process to converge fast. The HSA algorithm inherits the merit of the simulated annealing algorithm, and can avoid the disadvantage of blind search in the simulated annealing algorithm to some extent according to the special neighborhood. The computational results show that the performance of the presented algorithm outperforms that of the quasi-physical quasi-human algorithm.

1 Introduction

Given n objects, each with given shape and size, and a bounded space, the packing problem is how to pack the objects into the bounded space without overlapping. In this paper, we consider the circles packing problem, its aim is to pack different-sized circles into a larger circle. This problem has been shown to be NP-hard [1], investigations from the 70s of last century up to now, however, show that for NP-hard problems, there does not exist an algorithm that is both rigorous and fast [1,2]. Hence people turn to nature for wisdom, to obtain heuristic approximate algorithm that is not absolutely rigorous, but is of high speed, high reliability and high efficiency.

The packing problem arises in many scientific and engineering fields, and finds many significant practical applications. The packing problem has a long history in the literature [3,4,5,6]. But the history of the circles packing problem is very short [5,6], and some results can be found in [6]. For the simpler cases of the circles packing problem, it is possible to obtain optimal solutions based on lattice patterns. However, for more complex cases, we have to depend on fast heuristic algorithms to generate approximate solutions. At present, some heuristic algorithms have been presented. A fast quasi-physical algorithm has been presented in [7], its work path is to find natural phenomena in the physical world equivalent to the original mathematical problem, and then to observe the evolution of matter in it. However, due to the work path, this algorithm is too determinate, it is often easy to get stuck in local minimum. A very

clever polynomial time approximate algorithm has been reported in [2]. This algorithm is of great theoretical significance, but it is of less practical value. Several heuristic procedures have been developed in [9]. The heuristics are based on a variety of solution building rules that emulate a certain process of packing circles into a rectangle container. Unfortunately, the running time of this algorithm is not reported. Recently, an improved quasi-physical quasi-human (QPQH) algorithm has been given in [8]. This algorithm, which combines the quasi-physical approach and the quasi-human strategy, is one of the best current heuristic methods. However, for more complex problems, it is still difficult to jump out of local minimum and arrive at a nice place.

Inspired by biology evolution, physical process, social life etc., human beings have found many good algorithms, for example, genetic algorithm and simulated annealing algorithm (SA)[10]. Especially, SA is a general stochastic search algorithm for combinatorial optimization problems. In contrast to other local search algorithms, it provides more opportunities to escape from local minimum. However, it often costs too much time for finding a solution, this situation prevents it from being applied to many practical problems. What is more, for enhancing the performance of SA, some effective heuristic strategies are presented. These strategies are from nature, and can allow the iterative process to converge fast. Based on SA and heuristics presented by this paper, we present a heuristic simulated annealing algorithm (HSA) for the circles packing problem. HSA inherits the merit of SA, and can avoid the disadvantage of blind search in SA to some extent according to the special neighborhood. The computational results show that the performance of HSA outperforms that of QPQH.

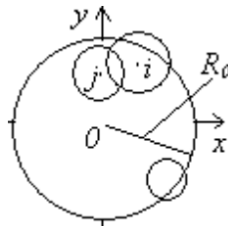


Fig. 1.

2 Mathematical Formulation of the Problem

Given an empty round plate and n circles of different sizes, here, n is a positive integer, we shall ask if these circles can be packed into the empty round plate without overlapping one another. This problem is stated more formally as follows [2,7,8]:

Take the origin of two-dimensional Cartesian coordinate system (See Fig. 1) at the central point of the round plate with the radius of R_0 . The coordinates of the center of circle i is denoted by (x_i, y_i) , the radius of circle i is R_i , we ask if there exist a set of real numbers or a configuration $(x_1, y_1, \dots, x_i, y_i, \dots, x_n, y_n)$, such that

$$\begin{cases} \sqrt{x_i^2 + y_i^2} \leq R_0 - R_i \\ \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq R_i + R_j \end{cases}, i, j = 1, 2, \dots, n, i \neq j.$$

If there exist such real numbers, then please give them.

According to elasticity mechanism, the extrusion elastic potential energy (u_{ij}) between two smooth elastic objects is proportional to the square of their mutual embedding depth

$$u_{ij} = kd_{ij}^2, i, j = 0, 1, \dots, n, i \neq j. \tag{1}$$

Here, k is a proportional coefficient, $k > 0$. In our computational program, we set $k = 1$. d_{ij} can be calculated as follows:

$$d_{0i} = \begin{cases} \sqrt{x_i^2 + y_i^2} + R_i - R_0 & \text{if } \sqrt{x_i^2 + y_i^2} + R_i > R_0, \\ 0, & \text{else,} \end{cases}$$

$$d_{ij} = \begin{cases} R_i + R_j - \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} & \text{if } \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} < R_i + R_j, \\ 0, & \text{else.} \end{cases}$$

According to (1), the extrusion elastic potential energy (U_j) possessed by circle j is

$$U_j = \sum_{i=0, j \neq i}^n u_{ij}, j = 1, 2, \dots, n.$$

Thus, the potential energy of the whole system is

$$U(X) = U(x_1, y_1, x_2, y_2, \dots, x_n, y_n) = \sum_{j=1}^n U_j. \tag{2}$$

Clearly, $U(X) \geq 0$. Hence the circles packing problem has been transformed into an optimization problem for the known potential energy function (2), that is to say, the minimum configuration $X^* = (x_1^*, y_1^*, \dots, x_n^*, y_n^*)$ of the potential energy function should be found. Therefore, if $U(X^*) = 0$, then X^* is one solution for the circles packing problem, whereas if $U(X^*) > 0$, then the problem has no solution.

3 The Proposed HSA Algorithm

3.1 Neighborhood Structure

Definition. For any given configuration $X = (x_1, y_1, \dots, x_i, y_i, \dots, x_n, y_n)$, we call $X' = (x_1, y_1, \dots, x'_i, y'_i, \dots, x_n, y_n)$ as a neighboring configuration of X , where the position of circle i in the configuration X' is randomly generated and different from the position of circle i in the configuration X , whereas the positions of other circles in X' is the same as X . The set of all these neighboring configurations of X is called the neighborhood of X , denoted by $N(X)$.

By Definition, there exist many configurations in $N(X)$ because circle i in X' is selected randomly and its position is generated randomly. Clearly, the range of search is too large, it can easily result to blind search and waste too much time, so the efficiency of computation is very low. According to these analyses, in order to make iterative process converge fast, it is necessary to reduce search space. Therefore, it is crucial to know how to generate X' for further constructing the neighborhood. In order to enhance the efficiency of the annealing process, according to the characteristics of the circles packing problem, we present the following heuristic strategy for generating X' by simulating circle's physical moving process: For any given $X = (x_1, y_1, \dots, x_i, y_i, \dots, x_n, y_n)$, $X' = (x_1, y_1, \dots, x'_i, y'_i, \dots, x_n, y_n)$ can be determined by the vector sum of embedding depth of circle i . Here, we look on the embedding depth as a vector. Thus, the displacement of circle i is the module of the vector sum of its embedding depth. The moving direction of circle i is the direction of the vector sum of the extrusion elastic forces acted on it. In Fig. 2, the way of generating X' is as follows.

For given $X = (x_1, y_1, \dots, x_i, y_i, \dots, x_n, y_n)$, we consider the change of circle i 's position under the extrusion elastic forces. With the help of Fig. 2, we have

$$\frac{x_i - x_j}{dx_{ij}} = \frac{D_{ij}}{d_{ij}}, \frac{y_i - y_j}{dy_{ij}} = \frac{D_{ij}}{d_{ij}}, \tag{3}$$

where, D_{ij} denotes the distance from the center of circle i to that of circle j , d_{ij} is the same as previous definition, dx_{ij} is the projection of d_{ij} in the horizontal axis x , dy_{ij} is the projection of d_{ij} in the vertical axis y . It is noted that dx_{ij}, dy_{ij} is not distance, so they may be positive or negative. From (3), we have

$$dx_{ij} = \frac{x_i - x_j}{D_{ij}} d_{ij},$$

$$dy_{ij} = \frac{y_i - y_j}{D_{ij}} d_{ij}.$$

Specially, we have

$$dx_{i0} = \frac{-x_i}{D_{0i}} d_{0i},$$

$$dy_{i0} = \frac{-y_i}{D_{0i}} d_{0i},$$

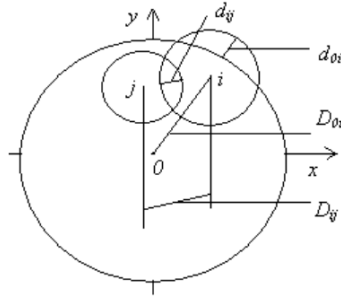


Fig. 2.

where, D_{0i} denotes the distance from the center of the plate to the center of circle i , d_{0i} is the same as previous definition, dx_{i0} is the projection of d_{0i} in the horizontal axis, dy_{i0} is the projection of d_{0i} in the vertical axis. Therefore, the next position of circle i is

$$\begin{cases} x'_i = x_i + dx_{ij} + dx_{i0} \\ y'_i = y_i + dy_{ij} + dy_{i0} \end{cases}, i = 1, 2, \dots, n.$$

With the help of the configuration $X = (x_1, y_1, \dots, x_i, y_i, \dots, x_n, y_n)$, we obtain a configuration $X' = (x_1, y_1, \dots, x'_i, y'_i, \dots, x_n, y_n)$ by simulating the physics moving process of circle i . It is noted that when we calculate X' , we only consider those circles or the plate that embed with circle i . Similarly, we can construct other neighboring configurations of X . Clearly, the number of the configurations in $N(X)$ is n . Since the way of generating X' is determinate, the range of search is significantly reduced. Therefore, we believe this strategy enlightened by physical process may avoid blind search and allow iterative process to converge fast.

3.2 Strategies of “Jumping out of the Trap”

For SA, when the temperature T tends to zero at the end of the process, the probability of accepting worse neighboring configurations is approximately zero. In that case, the SA algorithm loses its feature to accept worse configurations, thereby becoming identical to other local search algorithms. In this way the optimization process may get stuck in a local optimum. In addition, the way of generating X' may often lead the SA algorithm to get stuck in a local optimum during the course of execution. Under this circumstance, the promising approach is to put forward some good heuristic strategies for jumping out of the local minimum trap by taking the calculating

point out of the local minimum and place it in a position with better prospects. Then new SA process can be carried out over again.

This strategy of “jumping out of the trap” can be obtained by observing and learning from the social and nature phenomena and is therefore called heuristic strategy. In daily life, when we pack some objects into a trunk, we always pack large ones first, and then pack small ones. Otherwise, it is possible that the large objects cannot be packed into the trunk, because small ones have occupied the needed space of the large objects. Under this circumstance, we have to adjust the positions of small ones to find better layout. Namely, move away one small object from among small objects huddled together. What is more, the nature phenomena of “things of the same character repulse each other, things of the different character attract each other” are familiar. In addition, in order to decrease conflicts, the circle with maximum potential energy should be considered [8]. All of these inspire us to obtain the following heuristic strategies.

When getting stuck, we classify the circles by size. The circles of the same radius form one group, then we judge whether circles in the same group embed each other from the smallest group to the largest group in order. If circles in the same group embed each other, then randomly select an embedded circle and randomly determine its new position in the plate, and proceed with the SA process again. If circles in each group do not embed each other, then the circle with maximum potential energy can be picked out and randomly placed in the plate. These strategies are generally straightforward and intuitive, and can help the search process jump out of local minimum.

3.3 Statement of HSA

Suppose the circles have been sorted as M groups, from $i = 1$ to M , the radius increases in order. Integrating previous heuristic strategies into SA, HSA for the circles packing problem can be given as follows (See Fig. 3):

```

program HSA
  Begin
    Step one: Running the SA procedure: SA( );
    Step two: If  $U > 10^{-6}$  then
      Begin
         $i := 1$ ;
        While  $i \leq M$ 
          begin
            If circles in the  $i$ -th group embed each other;
              Begin;
                Randomly select an embedded circle in the  $i$ -
                th group and randomly determine its new position in
                the plate, go to Step one;
              End
             $i := i + 1$ ;
          end
        End
      End
  End

```

```

    The circle with maximum potential energy is
    picked out and randomly placed in the plate, go to Step
    one;
    End
    Step three. Stop.
end.

```

Fig. 3. The HSA algorithm

In Step two, we make use of strategies of “jump out of the trap”: dropping an overlapping circle in the group of small circles huddled together is to free space occupied by it and find better packing. In addition, dropping a circle with maximum potential energy is to decrease conflict. From statement of HSA, we understand that HSA includes SA, thus inherits the merit of SA.

4 Computational Results

We compare HSA with QPQH, both algorithms have been implemented with C language on an IBM PC-586 to perform large amounts of calculation with problem instances taken from [8] or generated for the purpose of this experiment. For the simpler cases of this problem, the calculations are carried out very smoothly and rapidly without exception. So what are chosen here are more complex cases. The 4 problem instances below, which include 1 instances of packing equal circles and 3 instances of packing unequal circles, are typical representatives. For each instance, 5 times of trial calculation have been executed with each algorithm. The average execution time, the times of reaching the solutions and the geometric morphologies of the solutions are shown in Table 1. Here, P denotes the problem instance, \bar{t} is used to denote the average execution time, other symbols is as before.

From the results, as shown in Table 1, we observed the speed of HSA is about eight times that of QPQH. For rather difficult problem instances, for example, instance 4, such an increase in speed is especially notable.

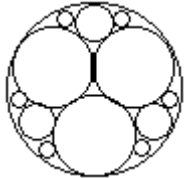
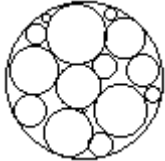
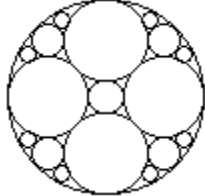
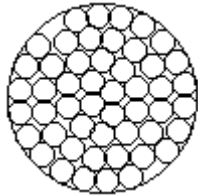
5 Conclusions

The heuristic strategies are from nature, so they are very intuitive, and can allow the iterative process to converge fast. HSA inherits the merit of SA, and can avoid the disadvantage of blind search in SA to some extent. The computational results show that the performance of HSA outperforms that of QPQH.

Owing to the complexity and intractability of the circles packing problem, the Hochbaum-Maass algorithm is impractical. This has been thoroughly analyzed by them [2]. Our algorithm is a direct answer to the question asked by Dorit S. Hochbaum and Wolfgang Maass. For the case of equal circles, some results can be found in [6,9,18], but the execution time is not reported. So HSA may remedy this

deficiency and is a significant supplement to this problem. Moreover, the global minimum configurations obtained by us have the same geometric morphologies as that obtained by [6,9,18].

Table 1. Comparisons between HSA and QPQH

P	R_0, n, R_1, \dots, R_n	HSA \bar{t}	QPQH \bar{t}	The geometric morphologies of the solutions
1	$R_0 = 215.47, n = 12,$ $R_1 = \dots = R_6 = 23.72,$ $R_7 = \dots = R_9 = 48.26,$ $R_{10} = R_{11} = R_{12} = 100$	9.412	422.114	
2	$R_0 = 39.37,$ $n = 15, R_1 = 1,$ $R_{i+1} = R_i + 1,$ $i = 1, 2, \dots, 14$	15.93	57.62	
3	$R_0 = 2.4143,$ $n = 17, R_{1-4} = 1,$ $R_{5-9} = 0.41415,$ $R_{10-17} = 0.2$	42.44	526.11	
4	$R_0 = 159.32,$ $n = 50,$ $R_{1-50} = 20$	168.27	439.72	

In addition, HSA is easily extendable to pack circles in other bounded space (for example, rectangle or triangle). The HSA algorithm may be of practical value to the rational layout of the round objects in the engineering fields, such as the optic-fiber communication and the transportation of the steel pipes in shipping containers. We hope to find highly efficient algorithm for other packing problem of even greater practical significance in the near future.

References

1. M.R. Garey, D.S. Johnson: *Computers and intractability: A Guide to the Theory of NP-Completeness*. San Francisco: Freeman (1979)
2. Dorit S. Hochbaum, Wolfgang Maass: Approximation schemes for covering and packing problems in image processing and VLSI, *Journal of the Association for Computing Machinery* 1(32) (1985) 130–136
3. Haessler. R.W. and Sweeney. P.E.: Cutting stock problems and solution procedures. *European Journal of Operational Research* 54 (1) (1991) 141–150
4. Dowsland. K.A., Dowsland. W.B.: Packing problems. *European Journal of Operational Research* 56 (1) (1992) 2–14
5. G. R. Raidl, G. Kodydek: Genetic Algorithms for the Multiple Container Packing Problem. in *Proc. of the 5th Int. Conference on Parallel Problem Solving from Nature V*, Amsterdam, The Netherlands, Springer LNCS (1998) 875–884
6. D. Lubachevsky, R.L. Graham: Curved hexagonal packing of equal disks in a circle. *Discrete & Computational Geometry* (1997) 179–194
7. Huang Wenqi, Zhan Shuhao: A quasi-physical method for solving packing problems, *Acta Mathematicae Applagatae Sinica* 2 (2) (1979) 176–180
8. Huaqing Wang, Wenqi Huang, Quan Zhang, Dongming Xu: An improved algorithm for the packing of unequal circles within a larger containing circle. *European Journal of Operational Research* 141 (2002) 440–453
9. John A.George, Jennifer M.George, Bruce W.Lammar: Packing different-sized circles into a rectangular container, *European Journal of Operational Research* 84 (1995) 693–712
10. S. Kirkpatrick, C. D. Gelatt, M.P. Vecchi: Optimization by simulated annealing. *Science* 220 (1983) 671–689