Chaos, Solitons and Fractals 89 (2016) 506–517

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

Quasi-human seniority-order algorithm for unequal circles packing

Dingju Zhu*

School of Computer Science, South China Normal University, Guangzhou 510631, China

ARTICLE INFO

Article history:
Received 31 January 2016
Accepted 23 February 2016
Available online 15 March 2016

Keywords:
Quasi-human
Seniority-order
Packing problem
Unequal circles
NPH problem

ABSTRACT

In the existing methods for solving unequal circles packing problems, the initial configuration is given arbitrarily or randomly, but the impact of different initial configurations for existing packing algorithm to the speed of existing packing algorithm solving unequal circles packing problems is very large. The quasi-human seniority-order algorithm proposed in this paper can generate a better initial configuration for existing packing algorithm to accelerate the speed of existing packing algorithm solving unequal circles packing problems. In experiments, the quasi-human seniority-order algorithm is applied to generate better initial configurations for quasi-physical elasticity methods to solve the unequal circles packing problems, and the experimental results show that the proposed quasi-human seniority-order algorithm can greatly improve the speed of solving the problem.

1. Introduction

Problem I (packing problem [1–6]): *N* different sizes and shapes of objects can be put down in a known container. The boundaries of the containers and objects are all rigid entities. If the objects do not fit objectively, we have to make a decision that they could not be put down into the container; otherwise, give each objects position and orientation.

Problem II (circle packing problem [1–10]): We simplify the packing problem. Only considering the circular containers and objects with the radius of a given container and the number and radius of objects to solve the problem whether there is a layout which makes containers and objects disjoint.

The quasi-human and quasi-physical method has been widely used and has achieved good results in solving the circles packing problems [1] and other NPH problems.

The circles packing problem [1–7] is a very important NPH problem. In the existing methods for solving unequal circles packing problems [1–3], the initial configuration is arbitrarily or randomly given, but, just as a good start is half the battle won, the impact of different initial configurations for existing packing algorithm to the speed of existing packing algorithm solving unequal circles packing problems is very large. The quasi-human seniority-order algorithm proposed in this paper can generate a better initial configuration to accelerate the speed of existing packing algorithm solving unequal circles packing problems. In experiments, the quasi-human seniority-order algorithm is applied to generate better initial configurations for the quasi-physical elasticity methods to solve the unequal circles packing problems, and the experimental results show that the proposed quasi-human seniority-order algorithm can greatly improve the speed of solving the problem. The existing quasi-physical elasticity methods for solving the unequal circles packing problems, in which an elastic squeeze between balls is simulated, are very simple but very effective; the balls move to where the pressure is smaller in the circular container, and then the configurations are gradually changed from the initial

* Tel.: +8618819181239.
E-mail address: zhudj@scnu.edu.cn

http://dx.doi.org/10.1016/j.chaos.2016.02.030
0960-0779/

configuration to the accepting or rejecting configuration. The existing quasi-physical elasticity methods are perfect in the algorithm, but the method has not optimized its initial configuration, which is given arbitrarily or randomly in the method. In the experiment, the initial configurations, which are arbitrarily and randomly given and generated by the proposed quasi-human seniority-order algorithm, are provided to the quasi-physical elasticity methods for solving the unequal circles packing problems, and the comparison of the results show that the proposed quasi-human seniority-order algorithm can greatly improve the speed of solving the problem. Moreover, we believe the quasi-human seniority-order algorithm can be applied to solve not only unequal circles packing problems but also other packing problems and complex problems [10,11] even NPH problems.

2. Quasi-human seniority-order (QS) algorithm

QS algorithm is not a packing algorithm, but an algorithm to generate a better initial configuration for existing packing algorithm to accelerate the speed of existing packing algorithm solving unequal circles packing problems.

2.1. Principle of QS algorithm

The purpose of QS algorithm is to generate a better initial configuration for existing packing algorithm to accelerate the speed of existing packing algorithm solving unequal circles packing problems. In order to reach this target, QS algorithm needs to follow the following principles:

First, when filling the container, the large objects should be selected in advance and then the small objects, so that the small objects fill the gaps among the large objects.

Second, the large object and the small object should be placed adjacently to avoid objects with the same or very similar size crowding together, for it is difficult to fill the gaps among adjacent objects with the same or very similar size with other objects.

Third, the small objects should be given opportunities to fill the central area of the container, for the large objects, which have filled the central area of the container, will block the small objects from entering the central area of the container, even if there are enough gaps among the large objects for the small objects.

2.2. Anthropomorphic definitions of QS algorithm

QS algorithm is an algorithm to generate a better initial configuration for existing packing algorithm, and a quasi-human algorithm, so we need to give the anthropomorphic definitions with the initial configuration.

The container of the packing problem is compared to a square. The objects that will fill the container are compared to persons who want to enter the square. The size of the objects is compared to the age of the persons.

There are m entrances around the square (the default value of m is 4). There is one VIP entrance in the center of the square. There is a separate queue for each entrance. The queue around the square usually is tangent to the square. The kth person in each queue is always younger than the front $k-1$ persons in any queue. This can be compared to the following configuration.

There are m entrances around the container (the default value of m is 4). There is one VIP entrance in the center of the container. There is a separate queue for each container. The queue around the container usually is tangent to the container. The kth object in each queue is always smaller than the front $k-1$ objects in any queue.

2.3. The steps of QS algorithm

Next, we will give the QS algorithm anthropomorphic steps.

Everyone who wants to enter the square should be lined up from oldest to youngest in a ready queue. If the age of a person in the ready queue is different from the age of the person in front of and behind him, then the person should be directly inserted into the VIP queue. There are m queues (the default value of m is 4). The 1st person out of those remaining in the ready queue should be inserted into the 1st queue; the 2nd person out of the those remaining in the ready queue should be inserted into the 2nd queue; the 3rd person out of those remaining in the ready queue should be inserted into the 3rd queue; the 4th person out of those remaining in the ready queue should be inserted into the 4th queue; ...; the mth person out of those remaining in the ready queue should be inserted into the mth queue. Then, the $(m+1)$th person out of those remaining in the ready queue should be inserted into the 1st queue; the $(m+2)$th person out of those remaining in the ready queue should be inserted into the 2nd queue; the $(m+3)$th person out of those remaining in the ready queue should be inserted into the 3rd queue; the $(m+4)$th person out of those remaining in the ready queue should be inserted into the 4th queue; ...; the $(m+m)$th person out of those remaining in the ready queue should be inserted into the mth queue. Then, just as before, the $(2m+1)$th person out of those remaining in the ready queue should be inserted into the 1st queue; the $(2m+2)$th person out of those remaining in the ready queue should be inserted into the 2nd queue; the $(2m+3)$th person out of those remaining in the ready queue should be inserted into the 3rd queue; the $(2m+4)$th person out of those remaining in the ready queue should be inserted into the 4th queue; ...; the $(2m+m)$th person out of those remaining in the ready queue should be inserted into the mth queue. This continues until there is no one remaining in the ready queue.

2.4. Special cases of QS algorithm

It should be noted that the statement “if the age of a person in the ready queue is different from the age of the person in front of and behind him, then the person should be directly inserted into the VIP queue” is given to avoid situation where the larger objects are crowded together and the smaller objects crowded together, which is consistent with the idea of the quasi-human seniority-order algorithm, namely, that “the large object and the small object should be placed adjacently to avoid objects with the same or very similar size crowding together”.