Optimal packing of 28 equal circles in a unit square – the first reliable solution∗

Mihály Csaba Markót

Institute of Informatics, University of Szeged, H-6720 Szeged, Árpád tér 2, Hungary

The paper deals with the problem class of finding the densest packings of non-overlapping equal circles within a unit square. We introduce a new interval branch-and-bound algorithm designed specifically for this optimization problem. After a brief description of the applied algorithmic tools, the capabilities of the algorithm are shown by solving the previously unsolved problem of packing 28 circles. The result confirms the optimality of an earlier found approximate solution and shows that it is unique in a certain sense.

Keywords: interval arithmetic, branch-and-bound methods, circle packing

AMS subject classification: 52C15, 52C26, 65G30, 90C30

1. Introduction

The original circle packing problem is the following: place a given number of equal circles without overlapping into a unit square maximizing the diameter of the circles as the objective function. It is quite easy to see that the problem of placing a given number of points into the unit square where the objective function to be maximized is the minimal distance between the pairs of points is equivalent to the circle packing one (see, e.g., [17]). Since the latter problem is easier to handle, in the sequel we deal with the point packing problem:

$$\max_{x, y} f_n(x, y), \quad \text{s.t.} \quad 0 \leq x_i, y_i \leq 1, \quad i = 1, 2, \ldots, n,$$

where $$f_n(x, y) = \min_{1 \leq i < j \leq n} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$, the unit square is $$[0, 1]^2$$, and the $$i$$th point is located at $$(x_i, y_i)$$. The number of the points, i.e. the integer $$n (\geq 2)$$ is considered as the parameter of the problem class.

In the last four decades several approaches were developed to solve circle packing problem instances. Until now, only the optimal packing of 2, 9, 14, 16, 25 and 36 circles are proved in a theoretical way. On the other hand, computer-aided optimality proofs exist for $$n \leq 20$$ [1,2,15] and for $$21 \leq n \leq 27$$ [14]. Recently, it was reported [8] that the function value of the currently known best packings are correct within the tolerance value of $$1e-5$$ for $$n = 10, \ldots, 35, 37, 38$$ (without determining the location of

* This work was supported by the Grants OTKA T 34350, OMFB D-30/2000 and OMFB E-24/2001.
all the optimizers). These methods are using the usual floating point arithmetic and they are bounding the rounding error only during the geometric steps of the algorithm. In contrast to that, the present paper introduces a fully interval-arithmetic based procedure providing the optimal values and all the possible optimizers with high accuracy.

The complete code and the running scripts of the algorithm are available at http://www.inf.u-szeged.hu/~markot/packcirc.htm.

2. Interval analysis and the branch-and-bound frame

The set of compact real intervals are denoted by \( \mathbb{I} \), where for all \( A \in \mathbb{I} \), \( A = [\underline{A}, \overline{A}] = \{a \in \mathbb{R} \mid \underline{A} \leq a \leq \overline{A}\} \). Here \( \underline{A}, \overline{A} \in \mathbb{R} \) mean the lower and upper bound of \( A \), respectively. The width of an interval is defined by \( w(A) = \overline{A} - \underline{A} \). A vector of \( n \) intervals is called an \( n \)-dimensional interval (or a box): \( X \in \mathbb{I}^n \). For a given set of points \( D \subseteq \mathbb{R}^n \), \( \mathbb{I}(D) \) denotes the set of all \( n \)-dimensional boxes \( X \) for which \( X \subseteq D \).

We call \( F: \mathbb{I}(D) \rightarrow \mathbb{I} \) an inclusion function of \( f: D \subseteq \mathbb{R}^n \rightarrow \mathbb{R} \), if \( f(X) = \{f(x) \mid x \in X\} \subseteq F(X) \) holds for all \( X \in \mathbb{I}(D) \), where \( f(X) \) is the range of \( f \) over \( X \). For more details on interval analysis, see [12].

We sketch the interval branch-and-bound algorithm [10,16] computing all the global maximizers of the general global optimization problem

\[
\max_{z \in Z_0} f(z),
\]

where \( f: \mathbb{R}^n \rightarrow \mathbb{R} \) is a continuous objective function and \( Z_0 \in \mathbb{I}^n \) is the search interval.

**Algorithm 1.**

1. \( Z := Z_0 \); WorkList := \( \{(Z, \overline{F}(Z))\} \); \( \tilde{f} := F(Z) \);
2. while (WorkList is not empty) do
3. \( (Z, \tilde{F}(Z)) := \text{Head(WorkList)} \);
4. Delete(Head(WorkList));
5. Bisection(\( Z, U^1, U^2 \));
6. for \( k := 1 \) to \( 2 \) do
7. Try to improve \( \tilde{f} \);
8. Use accelerating devices for \( U^k \);
9. if \( (U^k) \) is deleted as a whole) then continue with the next \( k \);
10. if \( w(F(U^k)) < \varepsilon \) then Insert(ResultList,\( (U^k, \overline{F}(U^k)) \));
11. else Insert(WorkList,\( (U^k, \overline{F}(U^k)) \));
12. Maximum := \( (\tilde{f}, \max(F(Z) \mid (Z, \overline{F}(Z)) \in \text{ResultList}) \));
13. return ResultList, Maximum;

In the following we specify some details of the algorithm:

An interval inclusion function \( F(Z) \) for the point packing problems. In [9] an appropriate inclusion function was discussed: