数理解析研究所講究録 第2108巻 2019年 43-49

正方形上への円充填問題に対するアルゴリズム
An Algorithm for Packing Circles in a Square

久野誉人，佐野良夫，渡邊雅弘
Takahito Kuno,Yoshio Sano,Masahiro Watanabe
筑波大学 システム情報工学研究科
Graduate School of Systems and Information Engineering
University of Tsukuba

2018 年 11 月 30 日

概要 円充填問題はコンテナ内に円を適切に配置する問題であり，産業分野への応用やヒューリスティクス解法によって規模の大きい問題が解けるようになったことで，近年様々な分野から注目されている．本研究では，正方形の中に複数の等しい大きさの円を充填する標準的な円充填問題に対して，厳密解を生成する分枝限定法について議論する．円の中心座標をそのものを扱うのではなく，座標対間の相対位置を考慮してモデル化したのち，アルゴリズムを構築し，実験結果を報告する．

1 はじめに

円充填問題とは，コンテナ内に n 個の円を重ならずに適切に配置する問題である．コンテナは正方形，長方形，円，立方体など様々な形が考えられている．充填する円も，等しいサイズのほか i 番目の円の半径が $r_i = \frac{1}{\sqrt{i}}$ のようにサイズが変わる場合も考えられている．また，円の重なりを許したり，コンテナからのはみ出しを許すといった条件の拡張を行うことで，実問題にも広く応用することができる．実問題への応用の具体例としては，UAV（無人航空機）が無線通信の基地局となり，地上のカバレッジエリアを考える問題がある．UAV に搭載されているアンテナには指向性があり，また同じ周波数帯の電波が重なると干渉が発生してしまうという条件のもと，円充填問題を利用している (図 1)[1]．その他にも，無線基地局の配置や，建物内の部屋や設備のレイアウト，貨物輸送コンテナへの最大積載，センサ・ロボット間の通信制御などへの応用がある [2]．計算機の性能向上によるヒューリスティクス解法の研究 [3] が進み，産業への応用が近年注目を集めている [4]．しかし，問題のわかりやすさに反して，円充填問題は厳密に解くことが非常に難しいことでも知られている [5]．最も簡単なクラスである「正方形内に半径の等しい円を充填する」という問題に対しても，充填する円の数が高々 10 程度までしか数学的に証明された解は知られておらず [6]，数十年に渡って様々なモデルやアルゴリズムも用いた証明の研究が進められている [6, 7, 8]．

産業分野への応用では，コンテナのサイズが複雑であったり，円の重なりやコンテナの境界からのはみ出しを許すなどの条件が加わり，厳密に解くことが難しいため，ヒューリスティクスでのアプローチが多くなされている．一定精度の近似解がリアルタイムに必要な場合にはヒューリスティクスは現実的なアプローチである．また，問題のモデリングが求解の効率や精度に大きく影響するため，ビリヤードのような完全弾

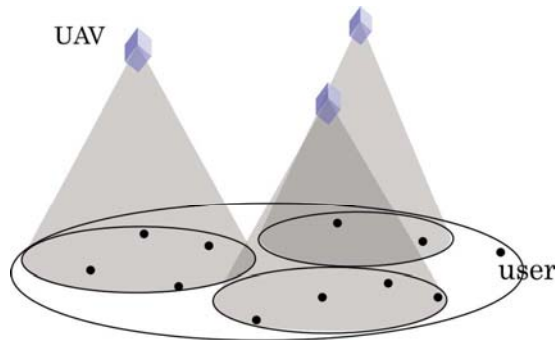

図 1: UAV(無人航空機) による無線カバレッジ問題 [1]

性衝突する円や，円の六角形への置き換え，円同士のエネルギー関数などを用いたモデルなどが考案されている [7].

円充填問題に対する厳密解法では，現時点で，正方形に対する円充填問題 (PACS) に対して円の数が $n \leq 30, n = 36$ の場合のみ一定の公差 ($\epsilon < 10^{-5}$) 以内で最適半径が求められている [6]．正方形領域を分割して領域を走査するため，円の数が増えると組合せの数が急増し，規模の大きい問題の厳密解の生成は難しい．そのため，Locatelli らは分枝限定法を用いて厳密解を求めるアプローチを提案している．このアプローチでは，初期暫定解として既知の最良値を利用し，$n \leq 38$ 程度の問題に対して新たな上界を与えている．しかし，大規模クラスタを用いて数ヶ月も計算に時間がかかっており，より効率の良い分枝操作，限定操作が求められている．

本論文では，正方形コンテナに半径の等しい n 個の円を詰める標準的な円充填問題に対して，厳密解を生成するための新しい分枝限定法を提案し，数値実験の結果を報告し，現状の課題を述べる．

2　提案するアルゴリズム

考察の対象は以下の問題である:

問題 P. 単位正方形 $U = [0, 1]^2$ に含まれる半径が等しく，重ならない n 個の円の最大半径を求めよ．

この問題は，次の等価な問題に置き換えることができる．

問題 Q. 単位正方形 U 内の点対間の最小距離を最大化するような，n 個の点の位置を求めよ．

この 2 つの問題について，問題 P の最大半径を $r^\star(n)$，問題 Q で得られる最適距離を $d^\star(n)$ とすると，以下の関係が成り立つ:

$$r^\star(n) = \frac{d^\star(n)}{2(d^\star(n) + 1)}$$

したがって，これ以降問題 Q について考察する．問題 Q を少し一般化し，長方形 $U_x \times U_y$ 内の問題を考える．円 i の中心座標を (x_i, y_i) とすると，以下のように定式化できる:

$$
\begin{aligned}
\max \quad & z \\
\text{s. t.} \quad & (x_i - x_j)^2 + (y_i - y_j)^2 \geq z \quad (1 \leq i < j \leq n) \\
& 0 \leq x_i \leq U_x, 0 \leq y_i \leq U_y \quad (i = 1, \ldots, n)
\end{aligned} \tag{1}
$$

ここで，U_x, U_y は長方形の幅と高さとする．この問題の難しさは，$\mathcal{O}(n^2)$ 本もの非凸 2 次制約を含んでいる点にあり，円の数が増えると現実的な時間で解けない．そのため，まず 2 次制約を 1 次近似した緩和問題を定義する．円 i, j の中心座標の差を表す変数 x_{ij}, y_{ij} を導入し，その存在区間をそれぞれ $[a_{ij}, b_{ij}], [c_{ij}, d_{ij}]$ と表す．これにより，式 (1) の部分問題は以下のように書き換えることができる:

$$
\begin{aligned}
\max \quad & z \\
\text{s. t.} \quad & \left.\begin{aligned} & x_{ij}^2 + y_{ij}^2 \geq z \\ & x_{ij} = x_i - x_j, \quad y_{ij} = y_i - y_j \\ & a_{ij} \leq x_{ij} \leq b_{ij}, \quad c_{ij} \leq y_{ij} \leq d_{ij} \end{aligned} \right\} (1 \leq i < j \leq n) \\
& 0 \leq x_i \leq U_x, \quad 0 \leq y_i \leq U_y \quad (i = 1, \ldots, n)
\end{aligned} \tag{2}
$$

非凸 2 次制約を緩和するため，2 次関数の区間 $[a_{ij}, b_{ij}], [c_{ij}, d_{ij}]$ における凹包絡関数を次のように定義する:

$$
\begin{aligned}
f_{ij}(x) &= (a_{ij} + b_{ij})x - a_{ij}b_{ij} \\
g_{ij}(y) &= (c_{ij} + d_{ij})y - c_{ij}d_{ij}
\end{aligned} \tag{3}
$$

命題 1. 式 (3) で定義した凹包絡関数に対して，以下が成り立つ:

$$
\begin{aligned}
f_{ij}(x) \geq x^2 &\Longleftrightarrow a_{ij} \leq x \leq b_{ij} \\
g_{ij}(y) \geq y^2 &\Longleftrightarrow c_{ij} \leq y \leq d_{ij}
\end{aligned}
$$

この凹包絡関数を利用することで，問題 (2) は，以下のように線形計画問題へ緩和される:

$$
\begin{aligned}
\max \quad & z \\
\text{s. t.} \quad & \left.\begin{aligned} & (a_{ij} + b_{ij})x_{ij} + (c_{ij} + d_{ij})y_{ij} - z \geq a_{ij}b_{ij} + c_{ij}d_{ij} \\ & x_{ij} = x_i - x_j, \quad y_{ij} = y_i - y_j \\ & a_{ij} \leq x_{ij} \leq b_{ij}, \quad c_{ij} \leq y_{ij} \leq d_{ij} \end{aligned} \right\} (1 \leq i < j \leq n) \\
& 0 \leq x_i \leq U_x, \quad 0 \leq y_i \leq U_y \quad (i = 1, \ldots, n)
\end{aligned} \tag{4}
$$

この緩和問題の最適解を $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{z})$ と表す．

命題 2. 問題 (2) の最適値は $\bar{z}$ によって上から抑えられる．

3 分枝限定法

この節で，問題 (4) によって得られる部分問題 (2) の最適値に対する上界値 $\bar{z}$ を用いて，解く必要のない部分問題を枝刈りする分枝限定法を構築する．

3.1 アルゴリズムの概要

1. 円 i, j の中心座標の差 x_{ij}, y_{ij} の存在区間 $[a_{ij}, b_{ij}], [c_{ij}, d_{ij}]$ に対して緩和問題 (4) の解 $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{z})$ を求める．
2. 暫定解 $(\boldsymbol{x}^\star, \boldsymbol{y}^\star, z^\star)$ に対して $\bar{z} > z^\star$ が成り立てば，$(\boldsymbol{x}^\star, \boldsymbol{y}^\star, z^\star) \leftarrow (\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{z})$ として暫定解を更新する．成り立たない場合は探索を打ち切る（限定操作）
3. 緩和解 $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{z})$ における 2 次関数と凹包絡関数との誤差が大きい円対を選択し，その座標差の存在区間の 1 つを 2 つに分割する（分枝操作）
4. 生成された 2 つの部分問題に対してステップ 1 から繰り返す．

3.2 分枝操作

ステップ 3 の分枝操作を詳しく説明しよう．まず，緩和解 $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{z})$ での 2 次関数と凹包絡関数との誤差が大きい座標のペアを以下の手順で求める:

$$\begin{aligned}(q, r) &\in \operatorname{argmax}\{f_{ij}(\bar{x}_{ij}) - \bar{x_{ij}}^2 | 1 \le i < j \le n\} \\ (s, t) &\in \operatorname{argmax}\{g_{ij}(\bar{y}_{ij}) - \bar{y_{ij}}^2 | 1 \le i < j \le n\}\end{aligned} \tag{5}$$

求めた x 座標のペア (q, r) に対して，$f_{qr}(\bar{x}_{qr}) - \bar{x}_{qr}^2 > \max\{0, g_{st}(\bar{y}_{st}) - \bar{y}_{st}^2\}$ が成立した場合，区間 $[a_{qr}, b_{qr}]$ を $[a_{qr}, \bar{x}_{qr}]$,$[\bar{x}_{qr}, b_{qr}]$ に分割して，2 つの子問題を生成する．同様に y 座標のペア (s, t) に対しても，$g_{st}(\bar{y}_{st}) - \bar{y}_{st}^2 > \max\{0, f_{qr}(\bar{x}_{qr}) - \bar{x}_{qr}^2\}$ が成立した場合，区間 $[c_{st}, d_{st}]$ を $[c_{st}, \bar{y}_{st}]$,$[\bar{y}_{st}, d_{st}]$ に分割する．

どちらの条件も成立しない場合，これ以降の部分問題には保持している暫定解 $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{z})$ より良い実行可能解は存在しないことがわかるので探索を打ち切る．

命題 3. 式 (5) から得られるペア $(q, r), (s, t)$ に対し，

$$\begin{aligned}f_{qr}(\bar{x}_{qr}) - \bar{x}_{qr}^2 &\le 0 \\ g_{st}(\bar{y}_{st}) - \bar{y}_{st}^2 &\le 0\end{aligned}$$

が成り立てば部分問題 (2) の最適値 $\bar{z}^\star$ は以下で与えられる:

$$\bar{z}^\star = \min\{\bar{x}_{ij}^2 + \bar{y}_{ij}^2 \mid 1 \le i < j \le n\}$$

4 数値実験

前節の分枝限定法を，1 回の分枝操作で生成される 2 つの部分問題の緩和解の良い方を深さ優先探索するようにプログラム化し，許容誤差を $\varepsilon (= 10^{-3})$ として終了条件が満たされるまで探索を繰り返す計算実験を行った（実験 1）．この条件では組合せの数が非常に多く，現実的な時間内で計算が終了しなかったため，実行時間を 600 秒に限定した場合に対しても実験を行った．実験環境は表 1 の通りである．

表 1: 実験環境

CPU	Intel Core i7-4820K @ 3.70GHz
メモリー	32GB
使用言語	C#
線形ソルバー	Google Optimization Tools

実験 1

実験結果を表 2 にまとめる．表の第 2 列にプログラムの出力 $z^\star$，第 3 列に既知の最適値 z_{opt}，第 4 列にはその比を記載する．

表 2: 実験 1

n	目的関数値 $z^\star$	最適値 z_{opt}	$z^\star/z_{opt}$
2	0.2928932	0.2928932	1
3	0.2539563	0.2543331	0.9983
4	0.25	0.25	1
5	0.2071067	0.2071067	1
6	0.1820186	0.1876806	0.9698
7	0.1666667	0.1744576	0.9553

充填する円の数が $n = 6$ の段階で 90 分を超えても実験が終了せず，分枝木を最後まで探索することができなかった．そのため，実験 2 では実行時間に制限をつけ，解の精度を比較する．

実験 2

実験結果を表 3 に示す．

表 3: 実験 2

n	目的関数値 $z^\star$	最適値 z_{opt}	$z^\star/z_{opt}$
2	0.2928932	0.2928932	1
3	0.2539563	0.2543331	0.9983
4	0.25	0.25	1
5	0.2071067	0.2071067	1
6	0.1781751	0.1876806	0.9494
7	0.1666667	0.1744576	0.9553
8	0.1463415	0.1705407	0.8581
9	0.125	0.1666667	0.7500
10	0.1082405	0.1482043	0.7303
11	0.0974212	0.1423992	0.6841
12	0.0983641	0.1399588	0.7028

充填する円の数が少ない場合では，実験 1 に近い値が算出されており，円の数 n が増えた場合でもこの分

枝限定法がヒューリスティクスアルゴリズムとしても利用できることが確認された．図 2，3 に，$n = 7$ の場合の既知の最適解と提案するアルゴリズムの出力結果を描画する．

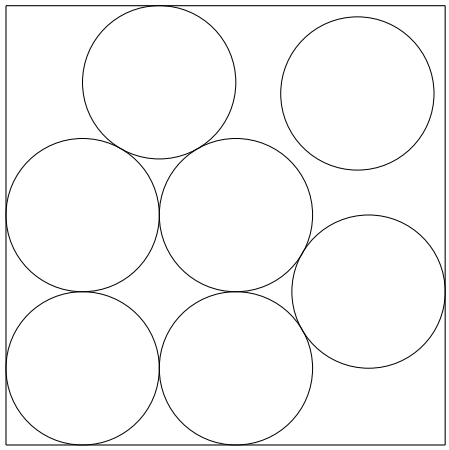

図 2: $n = 7$ に対しての最適解

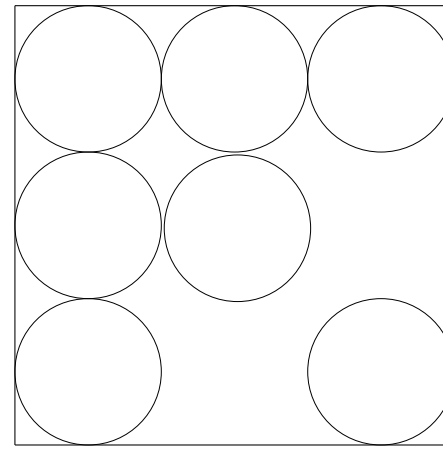

図 3: 提案するアルゴリズムで求めた解

5　今後の課題

正方形への円充填問題に対し，モデル化した問題の非凸 2 次制約を線形緩和し，その解を上界値に用いる分枝限定法を構築し，実装して数値実験を行った．実行時間を限定することで，この分枝限定法をヒューリスティクスとしても利用できることが確認できた．

実験 1 の結果より，分枝木をすべて探索した場合には円の数 n が 6 より大きくなった時点で，計算が現実的な時間内に収束しない．これは，円充填問題において円の配置が対称な場合 (図 4) や座標は同じであるがナンバリングのみ異なる場合 (図 5) があり，このアルゴリズムにおいてはその全てを繰り返し計算しているためだと考えられる．また，初期条件に対して，収束条件を満たすまで区間分割を行い繰り返し，部分問題を生成すると組合せの数が膨大になるため，あらかじめ初期解を点が分散した状態に設定することで，より効率的な求解ができるものと期待される．

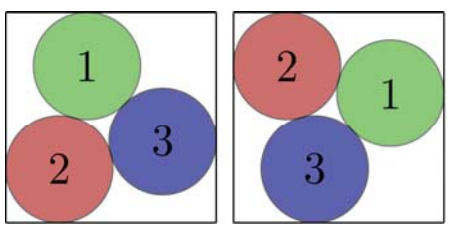

図 4: 円の配置が対称の場合

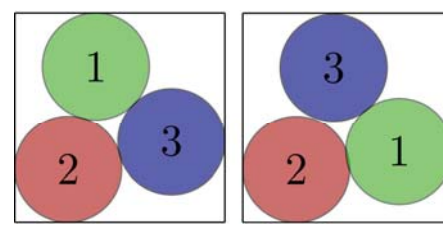

図 5: 円のナンバリングのみ異なる場合

実行時間を限定して，この分枝限定法をヒューリスティックとして利用する場合，区間の分割方法を工夫することで，さらに解の精度をあげられる可能性がある．今後は規模のより大きい問題や，解の収束性の改善を目指す．

参考文献

[1] M. Mohammad, S. Walid, B. Mehdi, and D. Merouane. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage. *IEEE Communications Letters*, Vol. 20, No. 8, pp. 1647–1650, 2016.

[2] M.C. Markót. Interval methods for verifying structural optimality of circle packing configurations in the unit square. *Journal of Computational and Applied Mathematics*, Vol. 199, pp. 353 – 357, 2003.

[3] C.O. López and J.E. Beasley. A heuristic for the circle packing problem with a variety of containers. *European Journal of Operational Research*, Vol. 214, No. 3, pp. 512 – 525, 2011.

[4] I. Castillo, F. J. Kampas, and J. D. Pintér. Solving circle packing problems by global optimization: Numerical results and industrial applications. *European Journal of Operational Research*, Vol. 191, No. 3, pp. 786 – 802, 2008.

[5] K. Stephenson. *Introduction to Circle Packing.* Cambridge University Press, 2004.

[6] E.Specht. Packomania. http://www.packomania.com.

[7] M. Locatelli and U. Raber. Packing equal circles in a square: I. theoretical results. *Technical Report 08-99, Dip. Sistemi e Informatica, Univ. di Firenze*, 1999.

[8] M. Locatelli and U. Raber. Packing equal circles in a square: a deterministic global optimization approach. *Discrete Applied Mathematics*, Vol. 122, No. 1, pp. 139 – 166, 2002.