axioms

MDPI

Article

Packing Spheres into a Minimum-Height Parabolic Container

Yuriy Stoyan [1], Georgiy Yaskov [1], Tetyana Romanova [1,2], Igor Litvinchev [3], José Manuel Velarde Cantú [4,*] and Mauricio López Acosta [4]

[1] Pidhornyi Institute of Mechanical Engineering Problems, vul. Komunalnykiv, 2/10, 61046 Kharkiv, Ukraine; stoyan@ipmach.kharkov.ua (Y.S.); yaskov@ipmach.kharkov.ua (G.Y.); t.romanova@leeds.ac.uk (T.R.)
[2] Leeds University Business School, University of Leeds, Leeds LS2 9JT, UK
[3] Graduate Program in Systems Engineering, Nuevo Leon State University (UANL), Av. Universidad s/n, Col. Ciudad Universitaria, San Nicolas de los Garza 66455, Mexico; igorlitvinchev@gmail.com
[4] Technological Institute of Sonora (ITSON), Navojoa-City 85870, Mexico; mlopeza@itson.edu.mx
* Correspondence: jose.velarde@itson.edu.mx

Abstract: Sphere packing consists of placing several spheres in a container without mutual overlapping. While packing into regular-shape containers is well explored, less attention is focused on containers with nonlinear boundaries, such as ellipsoids or paraboloids. Packing n-dimensional spheres into a minimum-height container bounded by a parabolic surface is formulated. The minimum allowable distances between spheres as well as between spheres and the container boundary are considered. A normalized Φ-function is used for analytical description of the containment constraints. A nonlinear programming model for the packing problem is provided. A solution algorithm based on the feasible directions approach and a decomposition technique is proposed. The computational results for problem instances with various space dimensions, different numbers of spheres and their radii, the minimal allowable distances and the parameters of the parabolic container are presented to demonstrate the efficiency of the proposed approach.

Keywords: packing; multidimensional spheres; paraboloid; Φ-function; nonlinear optimization

MSC: 05B40; 52C15; 52C17; 90C26

check for updates

Citation: Stoyan, Y.; Yaskov, G.; Romanova, T.; Litvinchev, I.; Velarde Cantú, J.M.; Acosta, M.L. Packing Spheres into a Minimum-Height Parabolic Container. *Axioms* **2024**, *13*, 396. https://doi.org/10.3390/axioms13060396

Academic Editors: Milena J. Petrović, Predrag S. Stanimirovic and Gradimir V. Milovanović

Received: 2 May 2024
Revised: 4 June 2024
Accepted: 7 June 2024
Published: 13 June 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Sphere packing is a well-studied area of research that involves arranging identical or non-identical spherical items within a given volume (container), subject to certain constraints. This problem has a wide range of applications, making it a versatile and important topic in both theoretical and practical contexts [1]. According to the typology of packing and cutting problems introduced in [2], packing problems are considered in two main formulations: open dimension problems (ODP) and knapsack problems. The ODP is aimed at optimizing the dimension(s) of a container, while packing the maximum number of identical objects or maximizing the total volume of packed objects is a knapsack problem. Typically, continuous nonlinear programming (NLP) models are used to formulate an ODP, while for the knapsack problem, mixed integer NLP models are implemented.

Our interest in irregular containers is motivated by the following considerations. Packing problems for regular-shaped containers (rectangles, circles) are well studied for 2D objects, such as circles [3], ovals [4,5] and ellipses [6–8]. Rich theoretical and empirical results are presented in these papers, where various NLP models and exact/heuristic/metaheuristic solution techniques are accompanied by extensive computational experiments to demonstrate the efficiency of the proposed approaches. Different NLP models and solution algorithms for packing 3D objects into regular 3D containers (cuboids, spheres, and cylinders) can be found, e.g., in [9,10] for spherical and in [11] for ellipsoidal shapes, together with corresponding empirical results obtained for different numbers of objects and various

container shapes. In all these works, geometric tools for modeling non-overlapping and containment conditions in Euclidean and non-Euclidean [4,5,9] metrics are provided. Algorithms based on combinations of smart heuristics and nonlinear optimization techniques are designed. The proposed solution approaches allow us to find the optimal solutions for small and medium-sized instances, while reasonably good feasible solutions are obtained for larger instances. However, as was highlighted in review papers [12–15], challenging packing problems in irregular containers are much less investigated. Several publications consider ellipsoidal containers [16,17], while there are only a few works focusing on packing for multiply connected domains and cardioids [18] and paraboloids [19]. Sphere packing into irregular containers arises in, e.g., material science [20,21] and nanotechnology [22]. A simple application of sphere packing into a parabolic container can be found in the food industry [23], where a parabolic dish container must be designed to store candies. To the best of our knowledge, in the n-dimensional case, the problem of packing spheres into an optimized parabolic container has not been considered before.

A brief review of the papers related to packing in irregular containers is provided below. An approach to constructing analytical non-intersection and containment conditions for non-oriented convex two-dimensional objects, defined by second-order curves, is proposed in [16]. This approach was applied to a problem of packing circles into an ellipse and minimizing the ellipse size. Packing algorithms applied to different-shaped two-dimensional domains are studied in [18], including rectangles, ellipses, crosses, multiply connected domains and even cardioid shapes. The authors introduce a novel approach centered around the concept of "image" disks, enabling the study of packing within fixed containers. Paper [19] focuses on Apollonian circle packing. The method has been used in various models, including geological sheer bands. Mathematical equations utilizing hyperbolas and ellipses are applied. This approach is applicable to a generic, closed, convex contour given the parametrization of its boundary. In the aerospace industry, packing into parabolic or other non-traditional containers is a significant challenge due to the specific shapes and delicate nature of many components [19]. The container is divided by horizontal racks into sub-containers. The proposed mathematical model considers the minimal and maximal allowable distances between objects subject to the behavior constraints of the mechanical system (equilibrium, moments of inertia and stability constraints). The paper describes a solution approach based on the multistart strategy, Shor's r-algorithm and accelerated search for the terminal nodes of the solution tree.

The objective of this paper is to develop a modeling and solutions approach to an ODP that consists of packing n-dimensional spheres into a minimum-height parabolic container. For analytical description of the placement constraints, the Φ-function technique [24] is used. This approach allows us to present mathematical models of optimized packing problems in the form of continuous NLP problems. To describe the containment of spheres in the parabolic container, a new Φ-function for the nD case is introduced. Using a section of the nD paraboloid and spheres by hyperplanes, it is iteratively reduced to consideration of the Φ-function in the 2D case. The Φ-function involves an additional variable parameter that is dynamically adjusted by solving a one-dimensional optimization problem. An approach based on the feasible directions method (FDM) [25] is developed considering the special properties of the Φ-function.

The contributions of the paper are as follows:

- A new problem of packing spheres into a minimum-height parabolic container in n-dimensional space;
- A new Φ-function for analytical description of the containment of a sphere into a parabolic container in n-dimensional space;
- An approach based on the feasible directions scheme considering the specific characteristics of the Φ-function.
- New benchmarks for various sphere radii and the parameters of the parabolic container in n-dimensional space for n = 2, 3, 4, 5.

The remainder of this paper is organized as follows. Section 2 describes the problem statement. Section 3 introduces geometrical tools for constructing a mathematical model of the packing problem. A mathematical model is formulated in Section 4. Section 5 presents a modification of the FDM. Section 6 provides the computational results for problem instances in several dimensions, with different numbers of spheres and their radii and various values of the minimal allowable distances and the parameters of the parabolic container. Section 7 concludes.

2. Problem Statement

Let a convex domain bounded by a parabolic surface and a hyperplane be defined in n-dimensional Euclidean space $\mathbb{R}^n$ as follows: $P_n(h) = P_n \cap H_n$ where $P_n = \{X = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n : \sum_{i=1}^{n-1} x_i^2 - 2px_n \leq 0\}$ and $H_n = \{X \in \mathbb{R}^n : x_n - h \leq 0\}$, i.e.,

$$P_n(h) = \{X \in \mathbb{R}^n : \sum_{i=1}^{n-1} x_i^2 - 2px_n \leq 0,\ x_n - h \leq 0\}. \quad (1)$$

Further, we refer the domain $P_n(h)$ to a container of variable height $h > 0$, with the predefined parameter $p > 0$.

And let a collection of nD spheres $S_j, j \in J = \{1, 2, \ldots, m\}$ with variable centers $y_j = (y_{j1}, y_{j2}, \ldots, y_{jn}) \in \mathbb{R}^n, j \in J$, be given and denoted by

$$S_j(y_j) = \{X \in \mathbb{R}^n : \|X - y_j\| - r_j \leq 0\}, j \in J. \quad (2)$$

In addition, the minimal allowable distances between each pair of spheres $S_t(y_t)$ and $S_j(y_j)$, as well as between a sphere $S_j(y_j)$ and the boundary of the container $P_n(h)$, are given, respectively, as δ_{tj} and δ_j.

Packing problem. Pack spheres $S_j(y_j)$, $j \in J$, into the minimum-height container $P_n(h)$, considering the minimal allowable distances δ_{tj}, $t < j \in J$, and δ_j, $j \in J$.

To describe the placement constraints of the packing problem analytically, the Φ-function technique is used. For the reader's convenience, the main definitions of the phi-function are provided in Appendix A. More details can be found in, e.g., Chapter 15 of [26].

The distance constraint for two spheres can be defined using the adjusted Φ-function in the form

$$\Phi_{tj}(y_t, y_j) = \|y_t - y_j\|^2 - (r_t + r_j + \delta_{tj})^2.$$

Therefore,

$$\Phi_{tj}(y_t, y_j) \geq 0 \Leftrightarrow dist(S_t(y_t), S_j(y_j)) \geq \delta_{tj},$$

$$dist(S_t(y_t), S_j(y_j)) = \min_{a \in S_t(y_t), b \in S_j(y_j)} \|a - b\|,\ a = (a_1, a_2, \ldots, a_n), b = (b_1, b_2, \ldots, b_n).$$

In Section 3, we introduce a continuous and everywhere defined function that allows us to describe the containment of each sphere into a parabolic container.

3. The Φ-Function for Containment Constraints

To describe analytically the containment constraint, $S_j(y_j) \subset P_n(h) \Leftrightarrow \text{int}S_j(y_j) \cap P_n^*(h) = \varnothing$, let us define a phi-function for an nD sphere $S_j(y_j)$ (2) and the object $P_n^*(h) = \mathbb{R}^n \backslash \text{int}P_n(h)$ (the compliment of the container $P_n(h)$ interior to the whole space $\mathbb{R}^n$).

Note that $P_n^*(h) = P_n^* \cup H_n^*$, where $H_n = \{X \in \mathbb{R}^n : x_n - h \geq 0\}$ and

$$P_n^* = \{X \in \mathbb{R}^n : \sum_{i=1}^{n-1} x_i^2 - 2px_n \geq 0\}. \quad (3)$$

A Φ-function for a sphere $S_j(y_j)$ and the object $P_n^*(h)$ can be stated in the following form:

$$\Phi_j^*(y_j, y_{jn}, h) \;=\; \min\{\; \Phi_j(y_j) - \delta_j, \Theta_j(y_{jn}, h)\}, \tag{4}$$

where $\Phi_j(y_j)$ is a Φ-function for a sphere $S_j(y_j)$ and the object P_n^*, and $\Theta_j(y_{jn}, h) \;=\; h - y_{jn} - r_j$ is a Φ-function for a sphere $S_j(y_j)$ and a half-space H_n^*.

Let us define a Φ-function for a sphere $S_j(y_j)$ and the object P_n^*. We state that $S_j(y_j) \subset P_n$ if $P_n^* \cap \text{int}S_j(y_j) \;=\; \varnothing$.

Firstly, construct an $(n-1)$D hyperplane K_1 passing through axis Ox_n of P_n and the center $y_j \;=\; (y_{j1}, y_{j2}, \ldots, y_{jn})$ of the nD sphere $S_j(y_j)$. Then, the section $K_1 \cap P_n$ yields the parabolic domain

$$P_{n-1} \;=\; \{X \in \mathbb{R}^{n-1} : \; -z_1^2 - \sum_{i\,=\,3}^{n-1} x_i^2 + 2px_n \;=\; 0\},$$

where $z_1 \;=\; \pm\sqrt{x_1^2 + x_2^2}$, while the section $K_1 \cap S_j(y_j)$ yields the $(n-1)$D-sphere $S_{j(n-1)}$ of radius r_j and the center $y_{j(n-1)} \;=\; (z_{j1}, y_{j3}, \ldots, y_{jn}), z_{j1} \;=\; \text{sign}(y_{j1})\sqrt{y_{j1}^2 + y_{j2}^2}$.

By analogy, an $(n-2)$D-hyperplane K_2 passing through axis Ox_n of P_{n-1} and the center $y_{j(n-1)}$ of $S_{j(n-1)}$ is constructed. Then, the section $K_2 \cap P_{n-1}$ yields the parabolic domain

$$P_{n-2} \;=\; \{X \in \mathbb{R}^{n-2} : -z_2^2 - \sum_{i\,=\,4}^{n-1} x_i^2 + 2px_n \;=\; 0\},$$

where $z_2 \;=\; \pm\sqrt{\sum_{i\,=\,1}^{3} x_i^2}$, while the section $K_2 \cap S_{j(n-1)}$ yields the $(n-2)$D-sphere $S_{j(n-2)}$ of radius r_j and the center $y_{j(n-2)} \;=\; (z_{j2}, y_{j4}, \ldots, y_{jn}), z_{j2} \;=\; \text{sign}(z_{j1})\sqrt{\sum_{i\,=\,1}^{3} y_{ji}^2}$.

The iterative procedure continues until the 2D parabolic domain $P_2 \;=\; \left\{X \in \mathbb{R}^2 : \; -z_{(n-1)}^2 + 2px_n \;=\; 0\right\}$ with $z_{n-1} \;=\; \text{sign}(z_{j(n-2)})\sqrt{\sum_{i\,=\,1}^{n-1} x_i^2}$ and the 2D sphere S_{j2} of radius r_j and the center $(z_{j(n-1)}, y_{jn})$ are obtained.

Note that $\text{sign}(z_{j(n-2)}) \;=\; \text{sign}(z_{j(n-3)}) \;=\; \ldots \;=\; \text{sign}(z_{j1}) \;=\; \text{sign}(y_{j1})$. This means that the construction of the Φ-function for the object P_n^* (3) and the nD sphere $S_j(y_j)$ (2) is reduced to deriving the Φ-function for the object P_2^* and the 2D sphere S_{j2}.

Let an equation of the tangent $03D2_j$ to the boundary of P_2 be given

$$f(z_{n-1}, x_n, t_j) \;=\; -z_{n-1}\sqrt{2p}t_j + p(x_n + t_j^2) \;=\; 0$$

for any $t_j \in \mathbb{R}^1$. Note that different tangents $03D2_j(t_j)$ can be generated for different values of t_j.

Let a point $(z_{n-1}, x_n) \;=\; (t_j\sqrt{2p}, t_j^2)$ be a tangency point of $03D2_j(t_j)$ and the boundary of P_2. Then, the normal equation of $03D2_j(t_j)$ takes the form

$$f_j(z_{n-1}, x_n, t_j) \;=\; -\frac{z_{n-1}\sqrt{2p}t_j - p(x_n + t_j^2)}{\sqrt{2pt_j^2 + p^2}} \;=\; 0. \tag{5}$$

Thus, the normalized Φ-function for the 2D sphere S_{j2} and the half-plane specified by the inequality $f_j(z_{n-1}, x_n, t_j) \leq 0$ can be defined as follows:

$$\Phi_{j0}(z_{j(n-1)}, y_{jn}, t_j) \;=\; f_j(z_{j(n-1)}, y_{jn}, t_j) - r_j. \tag{6}$$

Substituting $f_j(z_{n-1}, x_n, t_j)$ (5) into (6), the function $\omega_j(t_j) \;=\; (-z_{j,(n-1)}\sqrt{2p}t_j - p(y_{jn} + t_j^2))/\sqrt{2pt_j^2 + p^2}$ can be defined.

Then, we search for t_j^* at which the function $\omega_j(t_j^*)$ reaches the minimum corresponding to the distance between the center of S_{j2} and the boundary of P_2. Consequently, bearing in mind $z_{n-1} = \pm\sqrt{\sum_{i=1}^{n-1} x_i^2}$, the normalized Φ-function for $S_j(y_j)$ and P_2^* (3) takes the form

$$\Phi_j(y_j) = \min_{t_j \in [\beta_{j1}, \beta_{j2}]} \Phi_{j0}(z_{j(n-1)}, y_{jn}, t_j). \quad (7)$$

To find the optimal value of $t_j \in [\beta_{j1}, \beta_{j2}]$, a bisection technique [25] is applied.

Let us consider two cases for the locations of the center of S_{j2} with respect to P_2: case 1 corresponds to $(z_{j(n-1)}, y_{jn}) \in P_2$; case 2 corresponds to $(z_{j(n-1)}, y_{jn}) \notin P_2$.

Assume $(\widehat{z}_{j(n-1)}, \widehat{y}_{jn}) \in P_2$ and $\widehat{y}_{jn} \geq 0$. Here, $(\widehat{z}_{j(n-1)}, \widehat{y}_{jn})$ is the center point of the sphere S_{j2}. Let us consider two tangents $03D2_j(t_{j1})$ and $03D2_j(t_{j2})$ to $\mathrm{fr}P_2$ at points $A(\sqrt{2p\widehat{y}_{jn}}, \widehat{y}_{jn})$ and $B(\widehat{z}_{j(n-1)}, \widehat{z}^2_{j(n-1)}/(2p))$ for corresponding $t_{j1} = \sqrt{\widehat{y}_{jn}}$ and $t_{j2} = \widehat{z}_{j(n-1)}/\sqrt{2p}$ (Figure 1). Therefore, $[\beta_{j1}, \beta_{j2}] = [\widehat{z}_{j(n-1)}/\sqrt{2p}, \sqrt{\widehat{y}_{jn}}]$.

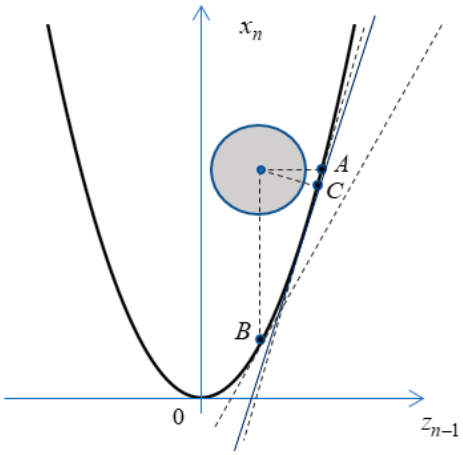


Figure 1. Illustration of interaction of S_{j2} and the boundary of P_2 in 2D.

In Figure 1, the segment AB of the parabola that corresponds to $t_j \in [\beta_{j1}, \beta_{j2}]$ is shown. The tangent $03D2_j(t_j^*)$ at the point $C(\sqrt{2p\widehat{y}_{jn}}, \widehat{y}_{jn}, t_j^*)$ corresponds to $t_j^* = \underset{t_j \in [\beta_{j1}, \beta_{j2}]}{\operatorname{argmin}} \Phi_{j0}(\widehat{z}_{j(n-1)}, \widehat{y}_{jn}, t_j)$.

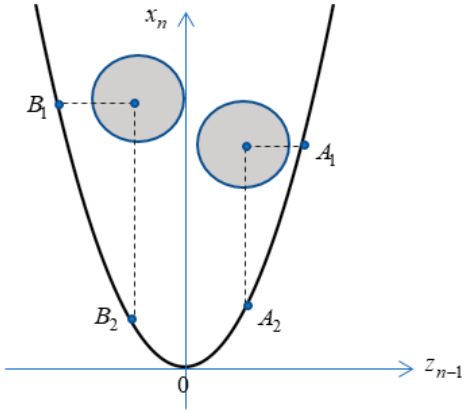
Let $(z_{j(n-1)}, y_{jn}) \in P_2$ (case 1); then,

$$[\beta_{j1}, \beta_{j2}] = \begin{cases} [\frac{z_{j(n-1)}}{\sqrt{2p}}, \sqrt{y_{jn}}] \text{ if } z_{j(n-1)} \geq 0 \text{ (case 1.1)} \\ [-\sqrt{y_{jn}}, \frac{z_{j(n-1)}}{\sqrt{2p}}] \text{ if } z_{j(n-1)} < 0 \text{ (case 1.2)} \end{cases}. \quad (8)$$

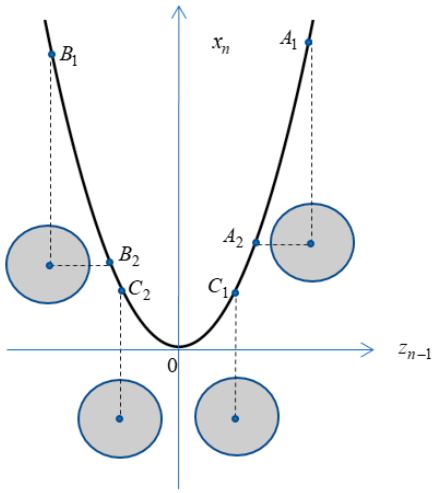
Let $(z_{j(n-1)}, y_{jn}) \notin P_2$(case 2); then,

$$[\beta_{j1}, \beta_{j2}] = \begin{cases} [\sqrt{y_{jn}}, \frac{z_{j(n-1)}}{\sqrt{2p}}] \text{ if } z_{j(n-1)} \geq 0,\ y_{jn} \geq 0 \text{ (case 2.1)} \\ [-\frac{|z_{j(n-1)}|}{\sqrt{2p}}, \frac{|z_{j(n-1)}|}{\sqrt{2p}}] \text{ if } y_{jn} < 0 \text{ (case 2.2)} \\ [-\frac{z_{j(n-1)}}{\sqrt{2p}}, \sqrt{y_{jn}}] \text{ if } z_{j(n-1)} < 0,\ y_{jn} \geq 0 \text{ (case 2.3)} \end{cases}. \quad (9)$$

Figure 2 illustrates two cases: a sphere is arranged inside the parabolic domain P_2 $\Phi_j(y_j) \geq 0$(Figure 2a), and a sphere is arranged outside the parabolic domain P_2, $\Phi_j(y_j) < 0$ (Figure 2b).



(**a**)



(**b**)

Figure 2. Arrangements of S_{j2} with respect to P_2 for different $t_j \in [\beta_{j1}, \beta_{j2}]$: (**a**) $(z_{j(n-1)}, y_{jn}) \in P_2$; (**b**) $(z_{j(n-1)}, y_{jn}) \notin P_2$.

In particular, to calculate $t \in [\beta_{j1}, \beta_{j2}]$, case 1.1 (8) is used for the parabolic segment A_1, A_2, while case 1.2 (8) is used for the parabolic segment B_1, B_2 (Figure 2a); case 2.1 (9) is used for the parabolic segment A_1, A_2, case 2.2 (9) is used for the parabolic segment C_1, C_2, and case 2.3 in (9) is used for the parabolic segment B_1, B_2 (Figure 2b).

4. Mathematical Model

A mathematical model of the packing problem can be formulated as follows:

$$\min_{Y,h} h \tag{10}$$

subject to

$$\begin{array}{rcl} \Phi_{tj}(y_t, y_j) & = & \|y_t - y_j\|^2 - (r_t + r_j + \delta_{tj})^2 \geq 0,\ t < j \in J \\ \Phi_j^*(y_j, y_{jn}, h) & = & \min\{\ \Phi_j(y_j) - \delta_j, \Theta_j(y_{jn}, h)\} \geq 0,\ j \in J \end{array} \tag{11}$$

where $Y = (y_1, y_2, \ldots, y_m) \in \mathbb{R}^{mn}$.

Note that the inequality $\Phi_j^*(y_j, y_{jn}, h) \geq 0$ in (11) is equivalent to the system of inequalities $\Phi_j(y_j) - \delta_j \geq 0$ and $\Theta_j(y_{jn}, h) \geq 0$ and ensures the arrangement of the nD sphere $S_j(y_j)$ fully inside the parabolic container $P_n(h)$.

The number of inequalities specifying the feasible region (11) is equal to $\chi = 0.5m(m-1) + 2m$. The dimensions of a solution matrix for W are $(mn+1) \times \chi$.

Thus, the number of inequalities/variables in the inequality system (11) is increased drastically by enlarging m. The solution matrix is strongly sparse. The problem is NP-hard [27].

Since we cannot define t_j explicitly, then we need to solve the optimization problems of the form (7) for each sphere $S_j(y_j)$ on each iteration of the solution process of the problem (10), (11). We do not use the solvers BARON [28] or IPOPT [29] for the original problem because of the dynamic nature of the Φ-function. Instead, a modification of the feasible directions method is developed.

The following solution strategy scheme for the problem (10), (11) is proposed:

1. Take a sufficiently large height h^0 of the container that guarantees a placement of spheres $S_j(y_j)$, $j \in J$, fully inside $P(h^0)$;
2. Generate the sphere centers $y_j^0, j \in J$, randomly so that $S_j(y_j^0) \subset P(h^0)$, $j \in J$, $\Phi_{tj}(y_t^0, y_j^0) \geq 0$, $t < j \in J$;
3. Apply the modification of the FDM to solve the problem (10), (11) for a set of feasible starting points.
4. Select the best solution.

5. Solution Algorithm

In contrast to the problem considered in [30], in this study, the spheres are of different radii, and the Φ-function describing the containment of a sphere into the container involves an additional parameter, which is dynamically changed during the optimization process.

The FDM for the problem (10), (11) is implemented using the iterative formula

$$Y^{k+1} = Y^k + \lambda^k Z^k, \; k = 0, 1, 2, \ldots, \tag{12}$$

where $Y^0 \in W$ (for $k = 0$) is a starting feasible point, Z^k is a search direction vector, and $\lambda^k > 0$ is a parameter that controls the step size.

Let $\varphi(Y) = h$ denote the objective function. A vector Z^k in (12) should provide $Y^k \in W$. To search for a vector Z^k, the following linear programming problem is solved:

$$(Z^k, \alpha^k) = \text{argmax}\alpha \text{ s.t. } 03D2 = (Z, \alpha) \in G^k, \tag{13}$$

$$\begin{aligned} G^k = \Big\{ & (Z, \alpha) \in \mathbb{R}^{\chi+1} : \nabla\Phi_{tj}(y_t^k, y_j^k) \cdot Z \geq \alpha, \; t < j \in J, \; \nabla\Phi_j(y_j^k) \cdot Z \geq \alpha, \\ & \nabla\Theta_j(y_{jn}^k, h^k) \cdot Z \geq \alpha, -\nabla\varphi(h^k) \cdot Z \geq \alpha, \; |z_i| \leq 1, \; i \in \Xi = \{1, 2, \ldots, \chi\}\}, \end{aligned} \tag{14}$$

where $Z = (z_1, z_2, \ldots, z_\chi) \in \mathbb{R}^\chi$, $\alpha \in \mathbb{R}^1$, $\Phi_j(y_j^k)$ is constructed according to (7).

Note that in (11), each $\Phi_{tj}(y_t, y_j)$ is an inverse convex function, and each $\Theta_j(y_{jn}, h)$ is a linear function. The vector of feasible directions may be orthogonal to the gradients of these constraints, so the inequalities $\nabla\Phi_{tj}(y_t^k, y_j^k) \cdot Z \geq \alpha$ and $\nabla\Theta_j(y_{jn}^k, h^k) \cdot Z \geq \alpha$ in (14) are replaced with $\nabla\Phi_{tj}(y_t^k, y_j^k) \cdot Z \geq 0$ and $\nabla\Theta_j(y_{jn}^k, h^k) \cdot Z \geq 0$, respectively.

To reduce the dimension of the problem (10), (11) and consequently of the problem (13), (14), a decomposition strategy [31] based on the degree of feasibility is employed.

When considering all the placed spheres, the majority of them are positioned at a significant distance from each other. The inequalities in the system (11) are satisfied with a considerable margin, and they can be disregarded when forming the optimization vector. At each step, only a subset of inequalities from the system (11) is considered, specifically those with a low degree of feasibility. During the optimization process, a parameter $\varepsilon^k > 0$ determining the degree of feasibility is adjusted dynamically, regulating the system constraints at each step. If an inequality is not considered in the optimization process at a certain step but is violated during the step's execution, the admissibility is controlled using the parameter λ^k in iterative Formula (12).

Let us denote the inverse convex and linear inequalities in the system (11) as $g_s(Y) \geq 0$, $s \in \Lambda \subset \Xi$ and the rest of the inequalities as $q_s(Y) \geq 0$, $s \in \Gamma \subset \Xi$ ($\Lambda \cup \Gamma = \Xi, \Lambda \cap$

$\Gamma = \varnothing$), and let $E^k = \left\{s \in \Lambda : 0 \le g_s(Y^k) \le \varepsilon^k\right\}$ and $B^k = \left\{s \in \Gamma : 0 \le q_s(Y^k) \le \varepsilon^k\right\}$. Here, $\varepsilon^k > 0$ is a threshold value. Then, the problem (13), (14) takes the form

$$(Z^k, \alpha^k) = \text{argmax}\alpha \text{ s.t. } (Z, \alpha) \in G_k, \tag{15}$$

$$\begin{aligned} G_k = \Big\{&(Z, \alpha) \in \mathbb{R}^{\chi+1} : \nabla g_i(Y^k) \cdot Z \ge 0,\ i \in E^k,\ \nabla q_i(Y^k) \cdot Z \ge \alpha,\ i \in B^k, \\ &-\nabla \varphi(h^k) \cdot Z \ge \alpha,\ |z_i| \le 1,\ i \in \Xi\Big\}. \end{aligned} \tag{16}$$

Taking into account the problem (15), (16), the following step-by-step algorithm is employed to solve problem (10), (11).

Step 1. Take a sufficiently large height h^0 of the container that guarantees a placement of spheres $S_j(y_j)$, $j \in J$, fully inside $P_n(h^0)$.

Step 2. Generate the sphere centers $y_j^0, j \in J$, randomly so that $S_j\left(y_j^0\right) \subset P_n\left(h^0\right)$, $j \in J, \Phi_{tj}(y_t^0, y_j^0) \ge 0,\ t < j \in J$.

Step 3. Set $k := 0, \varepsilon^0 := \varepsilon > 0$.

Step 4. Define the functions $\Phi_j(y_j^k)$ (7).

Step 5. Form the sets E^k, B^k.

Step 6. Set $\lambda^k := 1$.

Step 7. Calculate (Z^k, α^k) (Problem (15), (16)).

Step 8. If $\alpha^k \le 0$ (there is no a feasible direction decreasing the objective $\varphi(Y) = h$), then set $\varepsilon^k := \varepsilon^k/2$ and go to Step 5; otherwise ($\alpha^k > 0$), go to Step 9.

Step 9. Set $Y^{k+1} := Y^k + \lambda^k Z^k$ (12).

Step 10. If $Y^{k+1} \notin W$, then set $\lambda^k := \lambda^k/2$ and go to Step 9; otherwise, go to Step 11.

Step 11. If $\|Y^{k+1} - Y^k\| < \tau$, then stop algorithm; otherwise, set $\varepsilon^{k+1} := \varepsilon^k$, $k := k + 1$ and go to Step 4.

A schematic illustration of the proposed approach is shown in Figure 3. Three consecutive iterations of the FDM in 2D are illustrated in Figure 3a–c. An arrangement of 2D spheres corresponding to the stop criterion, $\|Y^{k+1} - Y^k\| < \tau$, at Step 11 is shown in Figure 3d.

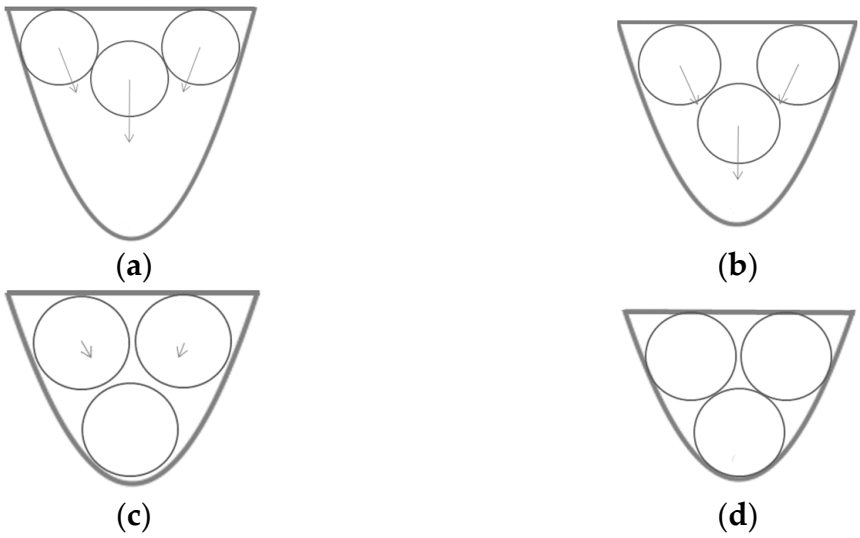


Figure 3. Illustration to the main stages of the solution procedure for three spheres: (**a**) an arrangement of spheres corresponding to the k-th iteration; (**b**) an arrangement of spheres corresponding to the $(k+1)$-th iteration; (**c**) an arrangement of spheres corresponding to the $(k+2)$-th iteration; (**d**) an arrangement of spheres corresponding to the stop criterion.

A flowchart corresponding to the solution strategy is presented in Figure 4.

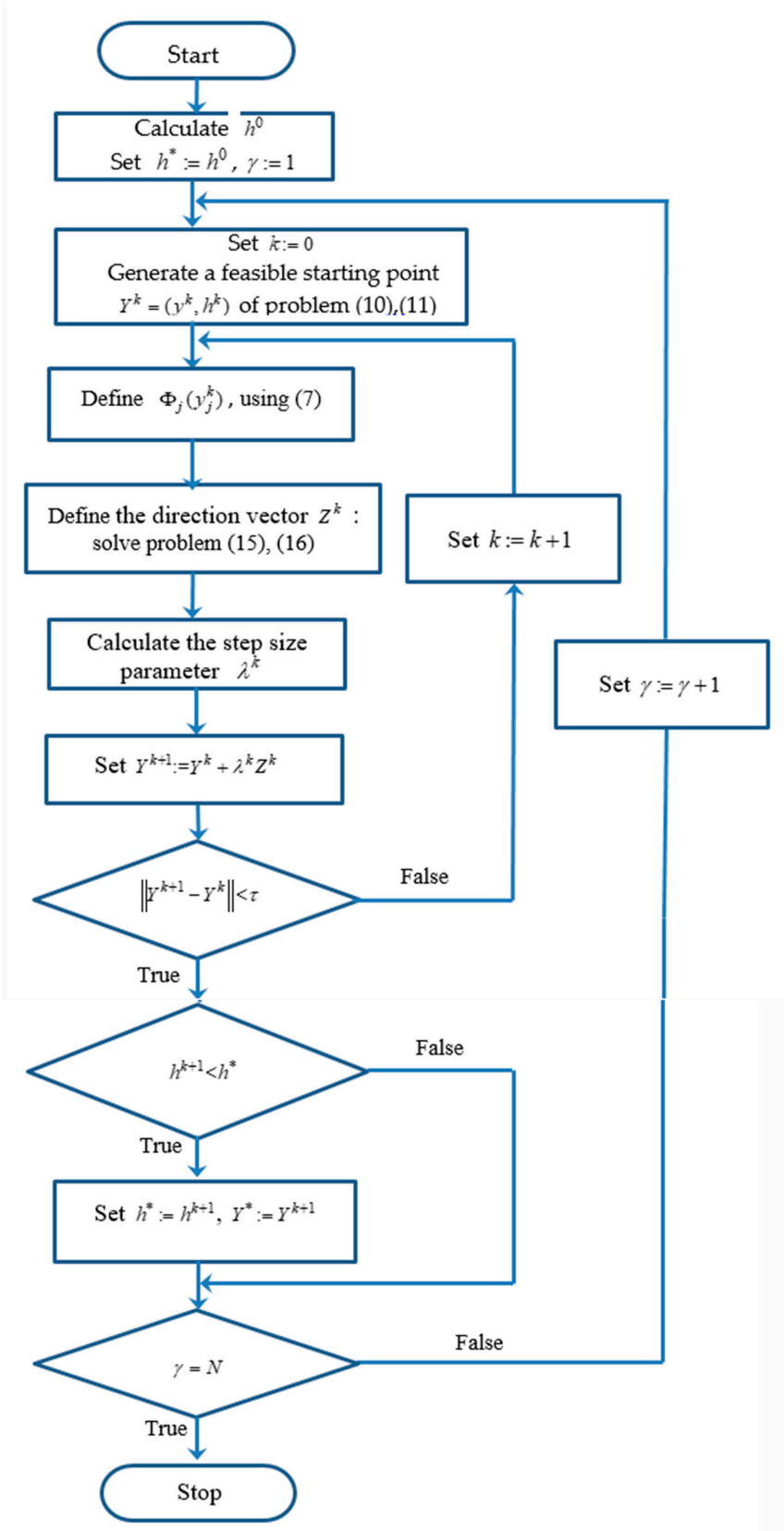


Figure 4. The flowchart of the main algorithm.

6. Computational Results

The proposed approach was numerically tested for eleven instances of the problem (9), (10) with different (a) dimensions, n = 2, 3, 4, 5; (b) numbers of spheres, m = 50, 100, 200; b) radii, $r_j, j = 1, \ldots, m$; (c) minimal allowable distances, δ_{tj}, δ_j, $t < j \in J$, $j \in J$; (d) parameters p of the parabolic container, $p = 1, 2, 5, 10$. For all the examples, we set $\varepsilon^0 = \varepsilon = 1, \tau = 10^{-6}$. For each problem instance, 10 starting points were generated. The computations were performed using an Intel® Core™ i3-6100T, 3.20 GHz, 8.00 GB of RAM.

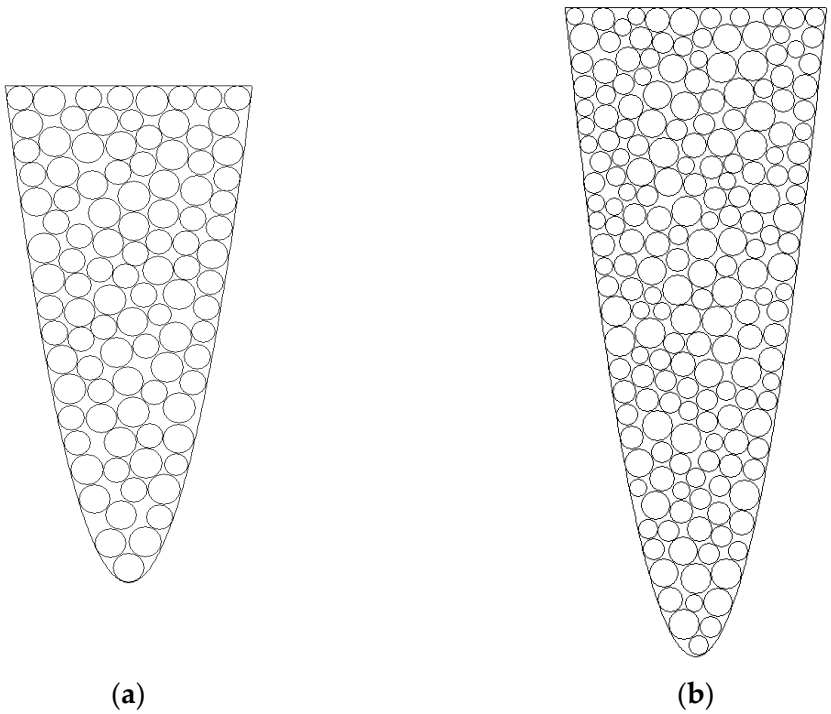
Example 1. $n = 2$, $m = 100$, $r_j = 1.177$, $j = 1, \ldots, 24$, $r_j = 1.117$, $i = 25, \ldots, 48$, $r_j = 0.97$, $j = 49, \ldots, 72$, $r_j = 0.927$, $j = 73, \ldots, 96$, $r_j = 0.86$, $j = 97, \ldots, 100$; $p = 1$, $\delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 5 min is $h* = 37.518079$.

Example 2. $n = 2$, $m = 200$, $r_j = 1.177$, $j = 1, \ldots, 24$, $r_j = 1.117$, $j = 25, \ldots, 48$, $r_j = 0.97$, $j = 49, \ldots, 72$, $r_j = 0.927$, $j = 73, \ldots, 96$, $r_j = 0.86$, $j = 97, \ldots, 120$, $r_j = 0.812$, $j = 121, \ldots, 144$, $r_j = 0.762$, $j = 145, \ldots, 168$, $r_j = 0.726$, $j = 169, \ldots, 192$, $r_j = 0.664$, $j = 193, \ldots, 200$; $p = 1$, $\delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 20 min is $h* = 49.511450$.

Example 3. $n = 2$, $m = 100$, the radii are as in Example 1; $p = 5$, $\delta_{tj} = \delta_j = 0$, $h^0 = 30$. The best solution found by our algorithm for 5 min is $h* = 21.695951$.

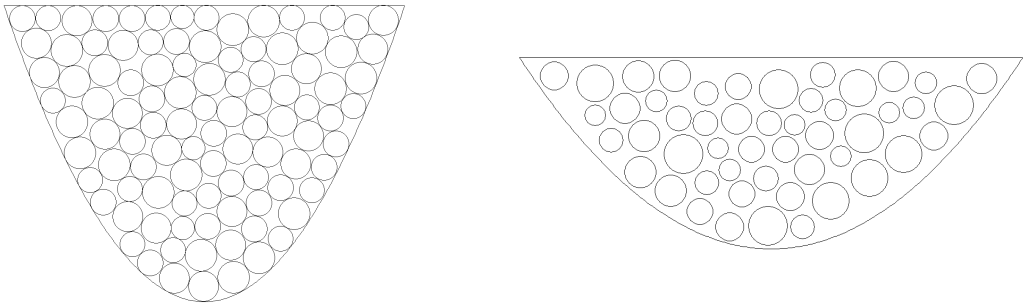
Example 4. $n = 2$, $m = 50$, $\{r_j, j = 1, \ldots, 50\}$ = {1.177, 1.177, 1.177, 1.177, 1.117, 1.117, 1.117, 1.117, 1.117, 0.970, 0.970, 0.970, 0.970, 0.970, 0.927, 0.927, 0.927, 0.927, 0.927, 0.860, 0.860, 0.860, 0.860, 0.860, 0.812, 0.812, 0.812, 0.812, 0.812, 0.762, 0.762, 0.762, 0.762, 0.762, 0.726, 0.726, 0.726, 0.726, 0.726, 0.664, 0.664, 0.664, 0.664, 0.664, 0.627, 0.627, 0.627, 0.627, 0.627}, $p = 5$, $\delta_{tj} \in [0.1, 0.5]$, $\delta_j \in [0.1, 1]$. The best solution found by our algorithm for 2 min is $h* = 11.577099$.

The corresponding placements of the spheres in Examples 1–4 are shown in Figure 5a–d.



(**a**) (**b**)

Figure 5. *Cont.*



(**c**) (**d**)

Figure 5. Optimized arrangement of 2D spheres: (**a**) Example 1; (**b**) Example 2; (**c**) Example 3; (**d**) Example 4.

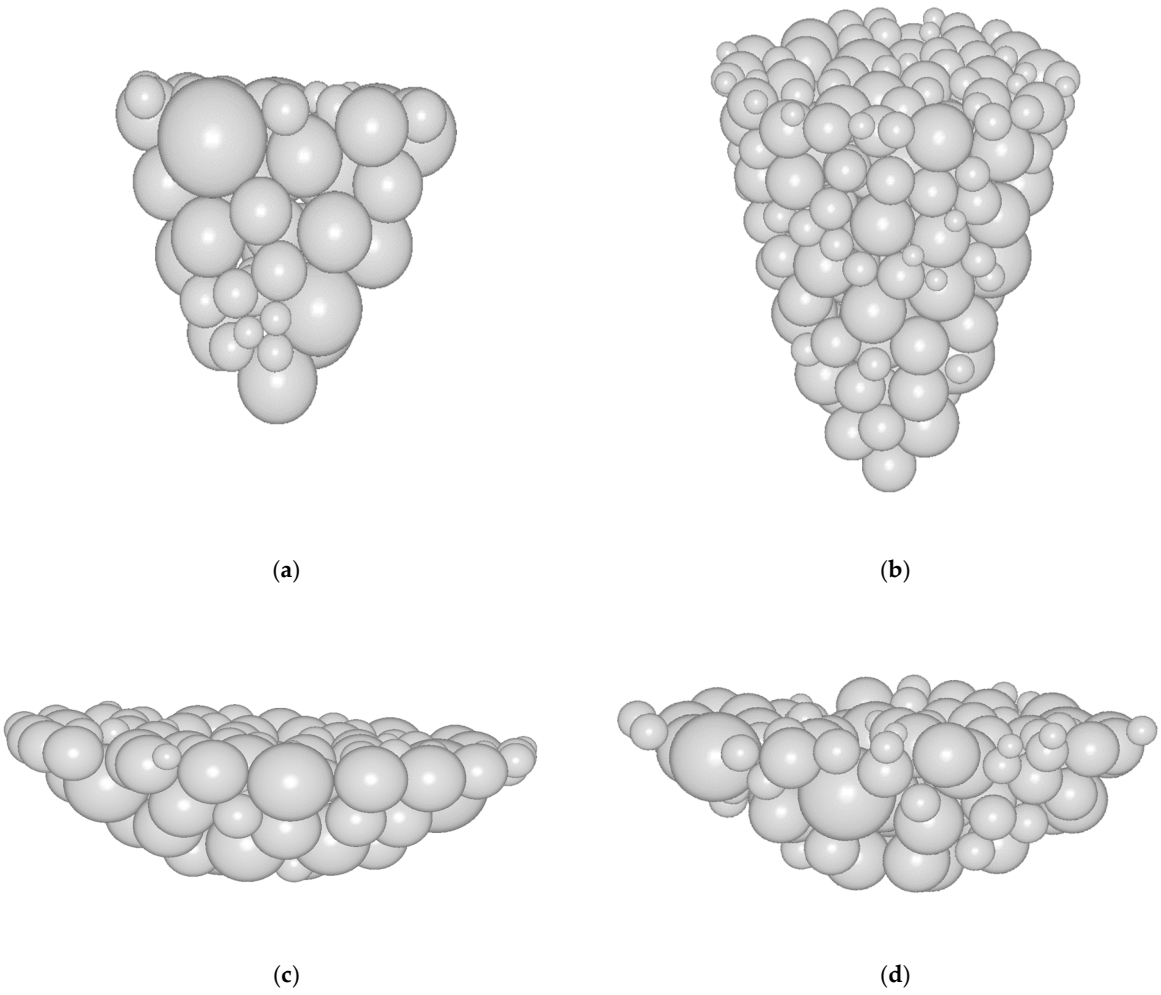
Example 5. $n = 3, m = 50, \{r_j, j = 1, \ldots, 50\} = \{0.527, 0.564, 0.566, 0.592, 0.612, 0.680, 0.747, 0.760, 0.807, 0.845, 0.850, 0.853, 0.855, 0.868, 0.887, 0.891, 0.934, 0.947, 0.955, 0.961, 1.044, 1.085, 1.180, 1.189, 1.210, 1.229, 1.237, 1.274, 1.275, 1.281, 1.292, 1.309, 1.325, 1.374, 1.399, 1.404, 1.430, 1.484, 1.491, 1.493, 1.525, 1.551, 1.551, 1.636, 1.670, 1.739, 1.819, 2.050, 2.171\}$, $p = 2, \delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 3 min is $h* = 11.927860$.

Example 6. $n = 3, m = 200, \{r_j, j = 1, \ldots, 200\} = \{2.171, 2.171, 2.171, 2.171, 2.050, 2.050, 2.050, 2.050, 1.819, 1.819, 1.819, 1.819, 1.739, 1.739, 1.739, 1.739, 1.670, 1.670, 1.670, 1.670, 1.636, 1.636, 1.636, 1.636, 1.551, 1.551, 1.551, 1.551, 1.551, 1.551, 1.551, 1.551, 1.525, 1.525, 1.525, 1.525, 1.493, 1.493, 1.493, 1.493, 1.491, 1.491, 1.491, 1.491, 1.484, 1.484, 1.484, 1.484, 1.484, 1.484, 1.484, 1.484, 1.430, 1.430, 1.430, 1.430, 1.404, 1.404, 1.404, 1.404, 1.399, 1.399, 1.399, 1.399, 1.374, 1.374, 1.374, 1.374, 1.325, 1.325, 1.325, 1.325, 1.309, 1.309, 1.309, 1.309, 1.292, 1.292, 1.292, 1.292, 1.281, 1.281, 1.281, 1.281, 1.275, 1.275, 1.275, 1.275, 1.274, 1.274, 1.274, 1.274, 1.237, 1.237, 1.237, 1.237, 1.229, 1.229, 1.229, 1.229, 1.210, 1.210, 1.210, 1.210, 1.189, 1.189, 1.189, 1.189, 1.180, 1.180, 1.180, 1.180, 1.085, 1.085, 1.085, 1.085, 1.044, 1.044, 1.044, 1.044, 0.961, 0.961, 0.961, 0.961, 0.955, 0.955, 0.955, 0.955, 0.947, 0.947, 0.947, 0.947, 0.934, 0.934, 0.934, 0.934, 0.891, 0.891, 0.891, 0.891, 0.887, 0.887, 0.887, 0.887, 0.868, 0.868, 0.868, 0.868, 0.855, 0.855, 0.855, 0.855, 0.853, 0.853, 0.853, 0.853, 0.850, 0.850, 0.850, 0.850, 0.845, 0.845, 0.845, 0.845, 0.807, 0.807, 0.807, 0.807, 0.760, 0.760, 0.760, 0.760, 0.747, 0.747, 0.747, 0.747, 0.680, 0.680, 0.680, 0.680, 0.612, 0.612, 0.612, 0.612, 0.592, 0.592, 0.592, 0.592, 0.566, 0.566, 0.566, 0.566, 0.564, 0.564, 0.564, 0.564, 0.527, 0.527, 0.527, 0.527\}$, $p = 2$, $\delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 35 min is $h* = 22.612047$.

Example 7. $n = 3, m = 100, \{r_j, j = 1, \ldots, 100\} = \{2.171, 2.171, 2.050, 2.050, 1.819, 1.819, 1.739, 1.739, 1.670, 1.670, 1.636, 1.636, 1.551, 1.551, 1.551, 1.551, 1.525, 1.525, 1.493, 1.493, 1.491, 1.491, 1.484, 1.484, 1.484, 1.484, 1.430, 1.430, 1.404, 1.404, 1.399, 1.399, 1.374, 1.374, 1.325, 1.325, 1.309, 1.309, 1.292, 1.292, 1.281, 1.281, 1.275, 1.275, 1.274, 1.274, 1.237, 1.237, 1.229, 1.229, 1.210, 1.210, 1.189, 1.189, 1.180, 1.180, 1.085, 1.085, 1.044, 1.044, 0.961, 0.961, 0.955, 0.955, 0.947, 0.947, 0.934, 0.934, 0.891, 0.891, 0.887, 0.887, 0.868, 0.868, 0.855, 0.855, 0.853, 0.853, 0.850, 0.850, 0.845, 0.845, 0.807, 0.807, 0.760, 0.760, 0.747, 0.747, 0.680, 0.680, 0.612, 0.612, 0.592, 0.592, 0.566, 0.566, 0.564, 0.564, 0.527, 0.527\}$, $p = 10, \delta_{tj} = \delta_j = 0$, $h^0 = 40$. The best solution found by our algorithm for 10 min is $h* = 7.577422$.

Example 8. $n = 3, m = 100$, the radii are as in Example 7; $p = 10, \delta_{tj} \in [0.1, 0.5]$, $\delta_j \in [0.1, 1]$. The best solution found by our algorithm for 10 min is $h* = 9.727001$.

The corresponding placements of the 3D spheres in Examples 5–8 are shown in Figure 6a–d.



(**a**) (**b**)

(**c**) (**d**)

Figure 6. Optimized arrangement of 3D spheres: (**a**) Example 5; (**b**) Example 6; (**c**) Example 7; (**d**) Example 8.

Example 9. $n = 4$, $m = 100$, the radii are as in Example 7; $p = 2$, $\delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 15 min is $h* = 22.612047$.

Example 10. $n = 4$, $m = 200$, the radii are as in Example 6; $p = 2$, $\delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 45 min is $h* = 14.628068$.

Example 11. $n = 4$, $m = 200$, the radii are as in Example 6; $p = 2$, $\delta_{tj} = \delta_j = 0$. The best solution found by our algorithm for 55 min is $h* = 11.791466$.

7. Conclusions

Employing mathematical models offers a structured and systematic approach to problem-solving, facilitating precise analysis and prediction of outcomes [32]. In this paper, a mathematical model for packing different spheres into a minimal-height parabolic container is proposed. Non-overlapping and containment conditions are formulated using the phi-function approach. The minimal allowed distance between the spheres and the boundary of the container is considered. The problem belongs to a class of irregular packing problems due to the nonstandard container shape.

To solve the corresponding nonlinear optimization problem, a feasible directions approach combined with the hot start technique is proposed. A decomposition scheme is applied to reduce the number of constraints in the subproblem used to find the search direction. Numerical experiments are provided to demonstrate the efficiency of the proposed solution scheme. A detailed description of the problem instances and corresponding solutions are reported to form a benchmark for future research.

Our future research is focused on the following issues. The number of non-overlapping constraints grows quadratically with an increase in the number of spheres, resulting in a large-scale optimization problem. These constraints have a specific structure which can be used either for direct solution of the original problem or to construct tight bounds for the optimal objective [33,34]. The proposed approach is based on modeling the interactions between the spheres and the boundary of the parabolic container. It also can be applied to a broader class of containers, e.g., circular hyperboloids (single- and double-sheeted), spheroids or ellipsoids. Packing problems on surfaces [35] can also be considered, as well as various applications of spherical systems [36] and logistics [37]. Some results in these directions are forthcoming.

Author Contributions: Conceptualization, G.Y., Y.S., T.R., I.L. and J.M.V.C.; methodology, G.Y., Y.S., T.R. and I.L.; software, Y.S. and M.L.A.; validation, G.Y. and J.M.V.C.; formal analysis, G.Y., Y.S., T.R. and I.L.; investigation, G.Y., Y.S., T.R., I.L., J.M.V.C. and M.L.A.; resources, G.Y. and Y.S.; data curation, G.Y. and Y.S.; writing—original draft preparation, G.Y., Y.S., T.R., I.L., J.M.V.C. and M.L.A.; writing, review and editing, G.Y., Y.S., T.R., I.L., J.M.V.C. and M.L.A.; visualization, G.Y. and Y.S.; supervision, G.Y., Y.S., T.R., I.L. and J.M.V.C.; project administration, J.M.V.C., M.L.A. and Y.S.; funding acquisition, J.M.V.C., M.L.A. and G.Y. All authors have read and agreed to the published version of the manuscript.

Funding: The second and the third authors were partially supported by the Volkswagen Foundation (grant #97775), and the third author was supported by the British Academy (grant #100072), while the last two authors were partially supported by the Technological Institute of Sonora (ITSON), Mexico, through the Research Promotion and Support Program (PROFAPI 2024).

Data Availability Statement: The data presented in this study are available on request from the corresponding authors.

Acknowledgments: The authors would like to thank anonymous referees for constructive and positive comments.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

For the reader's convenience, the main definitions and properties of the Φ-functions are provided. More details can be found, e.g., in [23,25], Chapter 15.

Let A be a geometric object. The position of the object A is defined by a motion vector $u_A = (v_A, \theta_A)$, where v_A is a translation vector and θ_A is a vector of the rotation parameters. The object A, rotated by θ_A and translated by v_A, is denoted by $A(u_A)$.

For two objects $A(u_A)$ and $B(u_B)$, a Φ-function allows us to distinguish the following three cases: (a) $A(u_A)$ and $B(u_B)$ do not overlap, i.e., $A(u_A)$ and $B(u_B)$ do not have any common points; (b) $A(u_A)$ and $B(u_B)$ are in contact, i.e., $A(u_A)$ and $B(u_B)$ have only common frontier points; (c) $A(u_A)$ and $B(u_B)$ are overlapping so that $A(u_A)$ and $B(u_B)$ have common interior points.

Following the definition [26], a continuous and everywhere defined function, denoted by $\Phi^{AB}(u_A, u_B)$, is called a Φ-function of the objects $A(u_A)$ and $B(u_B)$ if the following conditions are fulfilled:

$$\Phi^{AB}(u_A, u_B) > 0, \text{ for } A(u_A) \cap B(u_B) = \varnothing$$

$$\Phi^{AB}(u_A, u_B) = 0, \text{ for } \mathrm{int}A(u_A) \cap \mathrm{int}B(u_B) = \varnothing \text{ and } frA(u_A) \cap frB(u_B) \neq \varnothing;$$

$$\Phi^{AB}(u_A, u_B) < 0, \text{ for } \mathrm{int}A(u_A) \cap \mathrm{int}B(u_B) \neq \varnothing.$$

Here, frA denotes the boundary of the object A, while $\mathrm{int}A$ stands for its interior. Thus,

$$\Phi^{AB}(u_A, u_B) \geq 0 \Leftrightarrow \mathrm{int}A(u_A) \cap \mathrm{int}B(u_B) = \varnothing$$

To describe a containment constraint $A(u_A) \subset B(u_B)$, a phi-function for the objects A and $B^* = R^n \backslash \text{int}B$ is used.

In the case, $\Phi^{AB^*}(u_A, u_B) \geq 0 \Leftrightarrow \text{int}A(u_A) \cap \text{int}B^*(u_B) = \varnothing \Leftrightarrow A(u_A) \subset B(u_B)$.

To model the distance constraints for two objects, the normalized Φ-function is applied.

A Φ-function of the objects $A(u_A)$ and $B(u_B)$ is called a normalized phi-function $\widetilde{\Phi}^{AB}(u_A, u_B)$ if the values of the function coincide with the Euclidean distance between the objects $A(u_A)$ and $B(u_B)$ when $\text{int}A(u_A) \cap \text{int}B(u_B) = \varnothing$.

Therefore,

$$\widetilde{\Phi}^{AB}(u_A, u_B) \geq \rho \Leftrightarrow dist\{A(u_A), B(u_B)\} \geq \rho.$$

References

1. Scheithauer, G. Introduction to Cutting and Packing Optimization. In *International Series in Operations Research & Management Science*; Springer: Cham, Switzerland, 2018; Volume 263, pp. 385–405. [CrossRef]
2. Wäscher, G.; Haußner, H.; Schumann, H. An improved typology of cutting and packing problems. *Eur. J. Oper. Res.* **2007**, *183*, 1109–1130. [CrossRef]
3. Castillo, I.; Kampas, F.J.; Pinter, J.D. Solving circle packing problems by global optimization: Numerical results and industrial applications. *Eur. J. Oper. Res.* **2008**, *191*, 786–802. [CrossRef]
4. Kampas, F.J.; Castillo, I.; Pinter, J.D. Optimized ellipse packings in regular polygons. *Optim. Lett.* **2019**, *13*, 1583–1613. [CrossRef]
5. Kallrath, J.; Rebennack, S. Cutting ellipses from area-minimizing rectangles. *J. Glob. Optim.* **2014**, *59*, 405–437. [CrossRef]
6. Pankratov, A.; Romanova, T.; Litvinchev, I. Packing ellipses in an optimized rectangular container. *Wirel. Netw.* **2020**, *26*, 4869–4879. [CrossRef]
7. Kampas, F.J.; Pintér, J.D.; Castillo, I. Packing ovals in optimized regular polygons. *J. Glob. Optim.* **2020**, *77*, 175–196. [CrossRef]
8. Castillo, I.; Pintér, J.D.; Kampas, F.J. The boundary-to-boundary p-dispersion configuration problem with oval objects. *J. Oper. Res. Soc.* **2024**, 1–11. [CrossRef]
9. Elser, V. Packing spheres in high dimensions with moderate computational effort. *Phys. Rev. E* **2023**, *108*, 034117. [CrossRef] [PubMed]
10. Litvinchev, I.; Fischer, A.; Romanova, T.; Stetsyuk, P. A new class of irregular packing problems reducible to sphere packing in arbitrary norms. *Mathematics* **2024**, *12*, 935. [CrossRef]
11. Kallrath, J. Packing ellipsoids into volume-minimizing rectangular boxes. *J. Glob. Optim.* **2017**, *67*, 151–185. [CrossRef]
12. Leao, A.A.S.; Toledo, F.M.B.; Oliveira, J.F.; Carravilla, M.A.; Alvarez-Valdes, R. Irregular packing problems: A review of mathematical models. *Eur. J. Oper. Res.* **2020**, *282*, 803–822. [CrossRef]
13. Guo, B.; Zhang, Y.; Hu, J.; Li, J.; Wu, F.; Peng, Q.; Zhang, Q. Two-dimensional irregular packing problems: A review. *Front. Mech. Eng.* **2022**, *8*, 966691.
14. Rao, Y.; Luo, Q. Intelligent algorithms for irregular packing problem. In *Intelligent Algorithms for Packing and Cutting Problem*; Engineering Applications of Computational Methods; Springer: Singapore, 2022; Volume 10. [CrossRef]
15. Lamas-Fernandez, C.; Bennell, J.A.; Martinez-Sykora, A. Voxel-based solution Aapproaches to the three-dimensional irregular packing problem. *Oper. Res.* **2023**, *71*, 1298–1317. [CrossRef]
16. Gil, M.; Patsuk, V. Phi-functions for objects bounded by the second-order curves and their application to packing problems. In *Smart Technologies in Urban Engineering*; Arsenyeva, O., Romanova, T., Sukhonos, M., Tsegelnyk, Y., Eds.; STUE 2022, Lecture Notes in Networks and Systems; Springer: Cham, Switzerland, 2023; Volume 536. [CrossRef]
17. Santini, C.; Mangini, F.; Frezza, F. Apollonian Packing of Circles within Ellipses. *Algorithms* **2023**, *16*, 129. [CrossRef]
18. Amore, P.; De la Cruz, D.; Hernandez, V.; Rincon, I.; Zarate, U. Circle packing in arbitrary domains featured. *Phys. Fluids* **2023**, *35*, 127112. [CrossRef]
19. Kovalenko, A.A.; Romanova, T.E.; Stetsyuk, P.I. Balance Layout Problem for 3D-Objects: Mathematical Model and Solution Methods. *Cybern. Syst. Anal.* **2015**, *51*, 556–565. [CrossRef]
20. Burtseva, L.; Pestryakov, A.; Romero, R.; Valdez, B. Petranovskii. Some aspects of computer approaches to simulation of bimodal sphere packing in material engineering. *Adv. Mater. Res.* **2014**, *1040*, 585–591. [CrossRef]
21. Ungson, Y.; Burtseva, L.; Garcia-Curiel, E.R.; Valdez Salas, B.; Flores-Rios, B.L.; Werner, F.; Petranovskii, V. Filling of Irregular Channels with Round Cross-Section: Modeling Aspects to Study the Properties of Porous Materials. *Materials* **2018**, *11*, 1901. [CrossRef] [PubMed]
22. Burtseva, L.; Valdez Salas, B.; Romero, R.; Werner, F. Recent advances on modelling of structures of multi-component mixtures using a sphere packing approach. *Int. J. Nanotechnol.* **2016**, *13*, 44–59. [CrossRef]
23. Available online: https://olofly.com/product/huni-badger-parabolic-dish-container/ (accessed on 7 April 2023).
24. Chernov, N.; Stoyan, Y.; Romanova, T. Mathematical model and efficient algorithms for object packing problem. *Comput. Geom. Theory Appl.* **2010**, *43*, 535–553. [CrossRef]
25. Nocedal, J.; Wright, S.J. *Numerical Optimization*; Springer Series in Operations Research and Financial Engineering; Springer: New York, NY, USA, 2006.
26. Kallrath, J. *Business Optimization Using Mathematical Programming*; Springer: London, UK, 2021; ISBN 978-3-030-73237-0.

27. Chen, D. Sphere Packing Problem. In *Encyclopedia of Algorithms*; Kao, M.Y., Ed.; Springer: Boston, MA, USA, 2008. [CrossRef]
28. Sahinidis, N. BARON User Manual v. 2024.5.8. Available online: https://minlp.com/downloads/docs/baron%20manual.pdf (accessed on 8 May 2024).
29. IPOPT: Documentation. Available online: https://coin-or.github.io/Ipopt/ (accessed on 14 January 2023).
30. Stoyan, Y.; Yaskov, G. Packing congruent hyperspheres into a hypersphere. *J. Glob. Optim.* **2012**, *52*, 855–868. [CrossRef]
31. Romanova, T.; Stoyan, Y.; Pankratov, A.; Litvinchev, I.; Marmolejo, J.A. Decomposition algorithm for irregular placement problems. In *Intelligent Computing and Optimization, Proceedings of the 2nd International Conference on Intelligent Computing and Optimization 2019 (ICO 2019), Koh Samui, Thailand, 3–4 October 2019*; Intelligent Systems and Computing; Springer: Cham, Switzerland, 2019; Volume 1072, pp. 214–221.
32. Animasaun, I.L.; Shah, N.A.; Wakif, A.; Mahanthesh, B.; Sivaraj, R.; Koriko, O.K. *Ratio of Momentum Diffusivity to Thermal Diffusivity: Introduction, Meta-Analysis, and Scrutinization*; Chapman and Hall/CRC: New York, NY, USA, 2022. [CrossRef]
33. Litvinchev, I.S. Refinement of Lagrangian bounds in optimization problems. *Comput. Math. Math. Phys.* **2007**, *47*, 1101–1108. [CrossRef]
34. Litvinchev, I.; Rangel, S.; Saucedo, J. A Lagrangian bound for many-to-many assignment problems. *J. Comb. Optim.* **2010**, *19*, 241–257. [CrossRef]
35. Lai, X.; Yue, D.; Hao, J.K.; Glover, F.; Lü, Z. Iterated dynamic neighborhood search for packing equal circles on a sphere. *Comput. Oper. Res.* **2023**, *151*, 106121. [CrossRef]
36. Asadi Jafari, M.H.; Zarastvand, M.; Zhou, J. Doubly curved truss core composite shell system for broadband diffuse acoustic insulation. *J. Vib. Control* **2023**. [CrossRef]
37. Bulat, A.; Kiseleva, E.; Hart, L.; Prytomanova, O. Generalized Models of Logistics Problems and Approaches to Their Solution Based on the Synthesis of the Theory of Optimal Partitioning and Neuro-Fuzzy Technologies. In *System Analysis and Artificial Intelligence*; Studies in Computational Intelligence; Zgurovsky, M., Pankratova, N., Eds.; Springer: Cham, Switzerland, 2023; Volume 1107, pp. 355–376. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.