Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 4136430, 12 pages
https://doi.org/10.1155/2019/4136430

Research Article

Optimized Packing Clusters of Objects in a Rectangular Container

T. Romanova,[1] **A. Pankratov**,[1] **I. Litvinchev**,[2] **Yu. Pankratova**,[3] **and I. Urniaieva**[3]

[1] *Department of Mathematical Modeling and Optimal Design, Institute for Mechanical Engineering Problems of the National Academy of Sciences of Ukraine, Kharkiv, 61046, Ukraine*
[2] *Graduate Program in Systems Engineering, Nuevo Leon State University (UANL), Mexico*
[3] *Department of System Engineering, Kharkiv National University of Radio Electronics, Kharkiv, 61166, Ukraine*

Correspondence should be addressed to T. Romanova; sherom@kharkov.ua

Received 30 October 2018; Accepted 6 January 2019; Published 5 February 2019

Academic Editor: Hong-Yu Wu

A packing (layout) problem for a number of clusters (groups) composed of convex objects (e.g., circles, ellipses, or convex polygons) is considered. The clusters have to be packed into a given rectangular container subject to nonoverlapping between objects within a cluster. Each cluster is represented by the convex hull of objects that form the cluster. Two clusters are said to be nonoverlapping if their convex hulls do not overlap. A cluster is said to be entirely in the container if so is its convex hull. All objects in the cluster have the same shape (different sizes are allowed) and can be continuously translated and rotated. The objective of optimized packing is constructing a maximum sparse layout for clusters subject to nonoverlapping and containment conditions for clusters and objects. Here the term sparse means that clusters are sufficiently distant one from another. New quasi-phi-functions and phi-functions to describe analytically nonoverlapping, containment and distance constraints for clusters are introduced. The layout problem is then formulated as a nonlinear nonconvex continuous problem. A novel algorithm to search for locally optimal solutions is developed. Computational results are provided to demonstrate the efficiency of our approach. This research is motivated by a container-loading problem; however similar problems arise naturally in many other packing/cutting/clustering issues.

1. Introduction

A two-dimensional packing problem of objects in a container is NP-hard [1]. This problem is typical in logistics (transporting rolls of wallpaper, pipes, boxes, paint buckets, etc.,) and also has important applications in computer science, industrial engineering, manufacturing and production processes, healthcare, project portfolio selection, nanophysics, and agriculture (see, e.g., [2] and the references therein).

Various shapes of objects were studied. In particular, cutting and packing problems for ellipses are presented in [3–6], circular packing problems are investigated in [7–13], and packing problems for convex polygons are considered in [11–16]. Papers [17–21] are devoted to irregular packing involving arbitrary shaped objects.

In many cases the objects are not independent and have to be grouped in a number of certain clusters of nonoverlapping objects. This is typical, for example, for a container loading problem [22], where the objects in a large maritime container may form various clusters according to a type of objects (similar shapes, parts of the same machine). Similarly, clusters can be formed according to a supplier or a client (final destination) to facilitate loading/unloading the container.

While the composition of the cluster (number of objects and their shapes) is typically predefined, the overall shape of the cluster is frequently not specified. Bearing in mind a cluster as a number of objects placed in a flexible sack we define the shape of a cluster as a convex hull of the objects in the cluster. Note that the objects are nonoverlapping and the shape of the cluster (convex hull) depends on the layout of the objects in the cluster.

Throughout this paper it is assumed homogeneous clusters composed of the same shapes (different sizes are allowed). The number of clusters as well as the number of objects and their shapes and sizes is given. The shape of the cluster is represented by the convex hull of the objects in the cluster. A feasible clusters layout has to meet the following conditions: (a) objects in the cluster are mutually nonoverlapping and (b) clusters are mutually nonoverlapping and nonoverlapping with the complement of the rectangular container.

Concerning optimized packing of clusters, different objectives can be used. For example, we may look for the "densest" layout fixing one dimension of the rectangular container and minimizing the other subject to feasibility of the clusters layout. On the contrary, we may fix both dimensions of the container and look for a "sparsest" layout maximizing a certain "distance" between the clusters. This objective is motivated by the need of more space between clusters to facilitate access for their loading/unloading. In this paper the latter approach is used.

To the best of our knowledge the problem of optimized packing for clusters of objects was never considered before. The main contributions of the paper are (1) new formulation of the layout problem for clusters of objects; (2) new mathematical modeling tools for representing nonoverlapping and containment of clusters; (3) novel NLP model for optimized packing of clusters; and (4) new algorithm to get initial feasible solution to accelerate a local optimization procedure.

The paper is organized as follows. In Section 2 a layout problem for clusters of objects is formulated. Section 3 provides nonoverlapping and containment conditions for clusters obtained by phi-function technique. Mathematical model for the optimized layout problem is stated in Section 4. Solution algorithm and numerical experiments are presented in Sections 5 and 6, respectively. Conclusions in Section 7 complete this paper.

2. Problem Formulation

Let $\Omega = \{(x, y) \in R^2 : 0 \leq x \leq l, 0 \leq y \leq w\}$ be a rectangular domain (a container) and T_i, $i \in \{1, 2, \ldots, n\} = I_n$, be an ordered collection of *convex objects*. Each object T_i is defined by its metrical characteristics. In particular, an ellipse E_i is defined by its semiaxes a_i and b_i, a circle C_i is defined by its radius r_i, and a convex polygon K_i is defined by its vertices $p_{ik} = (p_{ik}^x, p_{ik}^y)$, $k = 1, \ldots, l_i$.

The position of the object T_i is characterized by the vector of variable placement parameters $u_i = (v_i, \theta_i)$, where $v_i = (x_i, y_i)$ is a translation vector and θ_i is a rotation angle. The center of the object coincides with the origin of its local coordinate system. Rotated by an angle θ_i and translated by a vector v_i an object T_i is defined as $T_i(u_i) = \{p \in R^2 : p = v_i + M(\theta_i) \cdot p^0, \forall p^0 \in T_i^0\}$, where T_i^0 denotes nontranslated and nonrotated object T_i and $M(\theta_i) = \begin{pmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{pmatrix}$ is a standard rotation matrix.

Remark 1. The position of a circle C_i is characterized by the motion vector $u_i = (v_i, 0) = (x_i, y_i, 0)$.

In this paper we define a *cluster of objects* (or simply a *cluster*) as a group of objects having the same shape. The ordered collection of objects T_i, $i = 1, \ldots, n$, is divided into N clusters $\Lambda_1 = \{T_1, \ldots, T_{n_1}\}$, $\Lambda_2 = \{T_{n_1+1}, \ldots, T_{n_2}\}, \ldots, \Lambda_N = \{T_{n_{N-1}+1}, \ldots, T_n\}$ with respect to index sets $\Xi_1 = \{1, \ldots, n_1\}$, and $\Xi_2 = \{n_1 + 1, \ldots, n_2\}, \ldots, \Xi_N = \{n_{N-1} + 1, \ldots, n\}$, $n = \sum_{q=1}^N m_q$, where $m_q = \mathrm{card}(\Xi_q)$, $q \in \{1, 2, \ldots, N\} = J_N$.

We introduce vectors of *placement parameters of clusters* $\Lambda_1, \Lambda_2, \ldots, \Lambda_N$ in the form $z_1 = (u_1, \ldots, u_{n_1})$, $z_2 = (u_{n_1+1}, \ldots, u_{n_2}), \ldots, z_N = (u_{n_{N-1}+1}, \ldots, u_n)$ and use notation $\Lambda_q(z_q)$ for a cluster Λ_q that involves moving objects $T_i(u_i)$, $i \in \Xi_q$.

With each cluster $\Lambda_q(z_q) = \bigcup_{i \in \Xi_q} T_i(u_i)$ we associate the convex hull of objects $T_i(u_i)$, $i \in \Xi_q$, denoted by $\widehat{\Lambda}_q(z_q) = conv\Lambda_q(z_q)$, $q \in J_N$.

Throughout this paper the following definitions are used: (1) two clusters $\Lambda_q(z_q)$ and $\Lambda_g(z_g)$ do not overlap each other if their convex hulls $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$ do not overlap, $q > g \in J_N$; (2) a cluster $\Lambda_q(z_q)$ belongs to a container Ω if its convex hull $\widehat{\Lambda}_q(z_q)$ belongs to a container Ω, $q \in I_N$.

Cluster Layout Problem CLP. Arrange the collection of *clusters* $\Lambda_q(z_q)$, $q \in J_N$, sufficiently distant one from another into a rectangular container Ω, such that

$$\operatorname{int} T_i(u_i) \cap \operatorname{int} T_j(u_j) = \varnothing, \quad \text{for } i > j,\ (i, j) \in \Sigma_q \times \Sigma_q,\ q \in J_N, \tag{1}$$

$$\operatorname{int} \widehat{\Lambda}_q(z_q) \cap \operatorname{int} \widehat{\Lambda}_g(z_g) = \varnothing, \quad \text{for } q > g \in J_N, \tag{2}$$

$$\widehat{\Lambda}_q(z_q) \subset \Omega \quad \text{for each } q \in I_N, \tag{3}$$

The first constraint assures nonoverlapping of objects within the same cluster and the second guarantees nonoverlapping of clusters, while the third relation presents containment conditions for clusters into Ω.

3. Tools of Mathematical Modeling

In this study, we use phi-function technique as a constructive tool for mathematical modeling of placement constraints within the field of Packing&Cutting (see, e.g., [3, 23–25]). For the reader's convenience we provide definitions of a phi-function (normalized phi-function) and a quasi-phi-function (normalized quasi-phi-function) in the appendix. In this section we introduce new tools to describe placement constraints (2) and (3) analytically.

3.1. Nonoverlapping of Objects within a Cluster. To describe relation (1) we use phi-functions and quasi-phi-functions for a pair of different shapes of convex objects.

Let us consider two objects $T_i(u_i)$ and $T_j(u_j)$ that belong to a cluster $\Lambda_q(z_q)$, $i > j$, $(i, j) \in \Sigma_q \times \Sigma_q$, $q \in J_N$.

According to the definition of a phi-function [23], we have that $\Phi^{T_i T_j}(u_i, u_j) \geq 0$ if and only if $\operatorname{int} T_i(u_i) \cap \operatorname{int} T_j(u_j) = \varnothing$.

From the main property of a quasi-phi-function [3] it follows that if $\Phi'^{T_i T_j}(u_i, u_j, u_{ij}) \geq 0$ for some u_{ij}, then $\operatorname{int} T_i(u_i) \cap \operatorname{int} T_j(u_j) = \varnothing$, where u_{ij} is a vector of extra variables.

In particular, we provide nonoverlapping tools for a pair of ellipses, circles, and convex polygons.

Quasi-Phi-Function for Two Ellipses. Let $E_i(u_i)$ and $E_j(u_j)$ be ellipses given by their semiaxes (a_i, b_i) and (a_j, b_j), with placement parameters $u_i = (x_i, y_i, \theta_i)$ and $u_j = (x_j, y_j, \theta_j)$.

A quasi-phi-function for $E_i(u_i)$ and $E_j(u_j)$ can be defined as

$$\begin{aligned}\Phi'^{E_iE_j}&\left(u_i, u_j, u_{ij} = \phi_{ij}\right)\\ &= x'_{ij}\left(v_i, \phi_{ij}\right) - x'_{ji}\left(v_j, \phi_{ij}\right)\\ &\quad - \left(d_{ij}\left(\theta_i, \phi_{ij}\right) + d_{ji}\left(\theta_j, \phi_{ij}\right)\right),\end{aligned} \tag{4}$$

where

$$\begin{aligned}x'_{ij}\left(v_i, \phi_{ij}\right) &= x_i \cos\phi_{ij} - y_i \sin\phi_{ij},\\ x'_{ji}\left(v_j, \phi_{ij}\right) &= x_j \cos\phi_{ij} - y_j \sin\phi_{ij},\\ d_{ij}\left(\theta_i, \phi_{ij}\right) &= \sqrt{b_i^2 + (a_i^2 - b_i^2)\cos^2\left(\theta_i - \phi_{ij}\right)},\\ d_{ji}\left(\theta_j, \phi_{ij}\right) &= \sqrt{b_j^2 + \left(a_j^2 - b_j^2\right)\cos^2\left(\theta_j - \phi_{ij}\right)},\end{aligned} \tag{5}$$

$\phi_{ij} \in R^1$ is an auxiliary variable (an angle between a straight line passing through the origin of the global coordinate system XOY and axis OX).

Phi-Function for Two Circles. Let $C_i(v_i)$ and $C_j(v_j)$ be circles given by their radii r_i and r_j with placement parameters $v_i = (x_i, y_i)$ and $v_j = (x_j, y_j)$.

A phi-function for $C(v_i)$ and $C_j(v_j)$ has the form [23]

$$\Phi^{C_iC_j}\left(v_i, v_j\right) = \left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 - \left(r_i + r_j\right)^2. \tag{6}$$

Quasi-Phi-Function for Two Convex Polygons. Let convex polygons $K_i(u_i)$ and $K_j(u_j)$ be defined by their vertices $p_{ik} = (p_{ik}^x, p_{ik}^y)$, $k = 1, \ldots, l_i$, $p_{jk} = (p_{jk}^x, p_{jk}^y)$, $k = 1, \ldots, l_j$, with placement parameters $u_i = (x_i, y_i, \theta_i)$ and $u_j = (x_j, y_j, \theta_j)$ and let $P(u_{ij}) = \{(x, y) : \mu_{ij} = \cos\phi_{ij} \cdot x + \sin\phi_{ij} \cdot y + \gamma_{ij} \le 0\}$ be a half plane.

Based on [26] a quasi-phi-function for $K_i(u_i)$ and $K_j(u_j)$ can be derived in the form

$$\begin{aligned}\Phi'^{K_iK_j}&\left(u_i, u_j, u_{ij}\right)\\ &= \min\left\{\Phi^{K_iP}\left(u_i, u_{ij}\right), \Phi^{K_jP^*}\left(u_j, u_{ij}\right)\right\},\end{aligned} \tag{7}$$

where $u_{ij} = (\phi_{ij}, \gamma_{ij}) \in R^2$ is a vector of auxiliary variables,

$$\Phi^{K_iP}\left(u_i, u_{ij}\right) = \min_{1\le k\le l_i} \mu_{ij}\left(p_{ik}\right) \tag{8}$$

is a phi-function for $K_i(u_i)$ and $P(u_{ij})$ and

$$\Phi^{K_jP^*}\left(u_j, u_{ij}\right) = \min_{1\le k\le l_j}\left(-\mu_{ij}\left(p_{jk}\right)\right) \tag{9}$$

is a phi-function for $K_j(u_j)$ and $P^*(u_{ij}) = R^2 \setminus \operatorname{int} P(u_{ij})$.

3.2. Nonoverlapping of Clusters. Let us describe analytically the placement constraint (2). To this aim we introduce a quasi-phi-function for two objects $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$.

Now we consider an extension of the phi-function technique to convex hulls of clusters of objects. Note that convex hull of objects in a cluster has variable shape and variable metrical characteristics that depend on variable placement parameters of objects in the cluster.

Let $P_{qg} = \{(x, y) : \mu_{qg}(x, y) = \cos\varphi_{qg} \cdot x + \sin\varphi_{qg} \cdot y + \gamma_{qg} \ge 0\}$ be a half plane with variable placement parameters $(\varphi_{qg}, \gamma_{qg})$.

Proposition 2. *A continuous and everywhere defined function*

$$\begin{aligned}\Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}&\left(z_q, z_g, \varphi_{qg}, \gamma_{qg}\right)\\ &= \min\left\{\Phi^{\widehat{\Lambda}_qP_{qg}}\left(z_q, \varphi_{qg}, \gamma_{qg}\right), \Phi^{\widehat{\Lambda}_gP^*_{qg}}\left(z_g, \varphi_{qg}, \gamma_{qg}\right)\right\}\end{aligned} \tag{10}$$

is a quasi-phi-function for $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$, where $\Phi^{\widehat{\Lambda}_qP_{qg}}(z_q, \varphi_{qg}, \gamma_{qg})$ is a phi-function of $\widehat{\Lambda}_q(z_q)$ and a half plane P_{qg}, $\Phi^{\widehat{\Lambda}_gP^*_{qg}}(z_g, \varphi_{qg}, \gamma_{qg})$ is a phi-function of $\widehat{\Lambda}_g(z_g)$ and the half plane $P^*_{qg} = R^2 \setminus \operatorname{int} P_{qg}$.

Proof. Based on the properties of a quasi-phi-function [3] defined for two convex objects we can conclude that $\Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) \ge 0$ if and only if there exists at least one separated line $L_{qg}\{(x, y) : \mu_{qg}(x, y) = 0\}$ for some φ_{qg}, γ_{qg} such that $\widehat{\Lambda}_q(z_q) \subset P_{qg}$ and $\widehat{\Lambda}_g(z_g) \subset P^*_{qg}$ and therefore $\widehat{\Lambda}_q(z_q) \cap \widehat{\Lambda}_g(z_g) = \varnothing$ for $q > g \in J_N$.

Next we define a phi-function of $\widehat{\Lambda}_q(z_q)$ and a half plane P_{qg} in (10) in the form

$$\begin{aligned}\Phi^{\widehat{\Lambda}_qP_{qg}}&\left(z_q, \varphi_{qg}, \gamma_{qg}\right)\\ &= \min\left\{\Phi^{T_iP_{qg}}\left(u_i, \varphi_{qg}, \gamma_{qg}\right), i \in \Xi_q\right\},\end{aligned} \tag{11}$$

where $\Phi^{T_iP_{qg}}(u_i, \varphi_{qg}, \gamma_{qg})$ is a phi-function of an object $T_i(u_i)$ and half plane P_{qg}, $i \in \Xi_q$.

It follows from the properties of the convex hull of $\Lambda_q(z_q) = \bigcup_{i\in\Xi_q} T_i(u_i)$ and characteristics of a separating line between two convex objects that

(1) $\widehat{\Lambda}_q(z_q) \cap P_{qg} = \varnothing \Longleftrightarrow \Lambda_q(z_q) \cap P_{qg} = \varnothing$ and therefore $T_i(u_i) \cap P_{qg} = \varnothing$, for all $i \in \Xi_q$. It implies that $\min\{\Phi^{T_iP_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\} > 0$.

(2) $\operatorname{int}\widehat{\Lambda}_q(z_q) \cap \operatorname{int} P_{qg} \ne \varnothing \Longleftrightarrow \operatorname{int}\Lambda_q(z_q) \cap \operatorname{int} P_{qg} \ne \varnothing$ and therefore there exists at least one object $T_i(u_i)$, such that $\operatorname{int} T_i(u_i) \cap \operatorname{int} P_{qg} \ne \varnothing$, $i \in \Xi_q$. It implies that $\min\{\Phi^{T_iP_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\} < 0$.

(3) $(\operatorname{int}\widehat{\Lambda}_q(z_q) \cap \operatorname{int} P_{qg} = \varnothing) \wedge (fr\,\widehat{\Lambda}_q(z_q) \cap frP_{qg} \ne \varnothing) \Longleftrightarrow (\operatorname{int}\Lambda_q(z_q) \cap \operatorname{int} P_{qg} = \varnothing) \wedge (fr\,\Lambda_q(z_q) \cap frP_{qg} \ne \varnothing)$ and therefore $\operatorname{int} T_i(u_i) \cap \operatorname{int} P_{qg} = \varnothing$, for all $i \in \Xi_q$, and there exists at least one object $T_i(u_i)$, such that $fr\,T_i(u_i) \cap frP_{qg} \ne \varnothing$, $i \in \Xi_q$. It implies that $\min\{\Phi^{T_iP_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\} = 0$.

Thus, the function $\Phi^{\widehat{\Lambda}_qP_{qg}}(z_q, \varphi_{qg}, \gamma_{qg})$ defined in (11) is a phi-function for $\widehat{\Lambda}_q$ and a half plane P_{qg}.

Similarly a phi-function of $\widehat{\Lambda}_g(z_g)$ and a half plane $P^*_{qg} = R^2 \setminus \operatorname{int} P_{qg}$ takes the form

$$\Phi^{\widehat{\Lambda}_g P^*_{qg}}\left(z_g, \varphi_{qg}, \gamma_{qg}\right) = \min\left\{\Phi^{T_j P^*_{qg}}\left(u_j, \varphi_{qg}, \gamma_{qg}\right), j \in \Xi_g\right\}, \tag{12}$$

where $\Phi^{T_j P^*_{qg}}(u_j, \varphi_{qg}, \gamma_{qg})$ is a phi-function of object $T_j(u_j)$, $j \in \Xi_g$, and the half plane P^*_{qg}.

Thus, $\Phi^{\widehat{\Lambda}_q P_{qg}}(z_q, \varphi_{qg}, \gamma_{qg}) \geq 0$ if and only if $\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}) \geq 0$ for each $i \in \Xi_q$.

Now we can conclude that $\max_{(\varphi_{qg}, \gamma_{qg})} \Phi'^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$ is a phi-function for $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$ and therefore $\Phi'^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$ is a quasi-phi-function for the objects.

It should be noted that a quasi-phi-function

$$\Phi'^{\widehat{\Lambda}_q \widehat{\Lambda}_g}\left(z_q, z_g, \varphi_{qg}, \gamma_{qg}\right) - 0.5\rho \tag{13}$$

is a normalized quasi-phi-function for $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$ (see the appendix for more details). It means that if $\rho > 0$ is the distance between $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$, defined in the standard way as a minimum Euclidean pointwise distance, then the inequality $\Phi'^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) - 0.5\rho \geq 0$ implies $\operatorname{dist}(\widehat{\Lambda}_q(z_q), \widehat{\Lambda}_g(z_g)) \geq \rho$. □

Remark 3. We do not construct convex hulls of clusters to describe nonoverlapping (2) in our problem.

Further we provide nonoverlapping tools (phi-functions) for a half plane $P_{qg} = \{(x, y) : \mu_{qg}(x, y) = \cos\varphi_{qg} \cdot x + \sin\varphi_{qg} \cdot y + \gamma_{qg} \geq 0\}$ and the following types of objects $T_i(u_i)$, $i \in \Xi_q$: an ellipse $E_i(u_i)$; a circle $C_i(v_i)$; and a convex polygon $K_i(u_i)$.

Phi-Function for an Ellipse and Half Plane P_{qg}. A phi-function for an ellipse $E_i(u_i)$ given by its semiaxes (a_i, b_i) and half plane P_{qg} can be defined in the form:

$$\begin{aligned}\Phi^{E_i P_{qg}}\left(u_i, \varphi_{qg}, \gamma_{qg}\right) &= x_i \cos\varphi_{qg} + y_i \sin\varphi_{qg} + \gamma_{qg} \\ &\quad - \sqrt{\left(a_i^2 - b_i^2\right) \cdot \cos^2\left(\theta_i + \varphi_{qg}\right) + b_i^2}.\end{aligned} \tag{14}$$

Phi-Function for a Circle and Half Plane P_{qg}. The phi-function for a circle $C_i(v_i)$ given by its radius r_i and half plane P_{qg} is defined in [24] and has the form

$$\Phi^{C_i P_{qg}}\left(v_i, \varphi_{qg}, \gamma_{qg}\right) = x_i \cos\varphi_{qg} + y_i \sin\varphi_{qg} + \gamma_{qg} - r_i. \tag{15}$$

Phi-Function for a Convex Polygon and Half Plane P_{qg}. The phi-function for a convex polygon $K_i(u_i)$ given by its vertices $p_{ik} = (p^x_{ik}, p^y_{ik})$, $k = 1, \ldots, l_i$, and half plane P_{qg} is defined in [24] and has the form:

$$\Phi^{K_i P_{qg}}\left(u_i, \varphi_{qg}, \gamma_{qg}\right) = \min_{k=1,\ldots,l_i}\left(p^x_{ik}\cos\varphi_{qg} + p^y_{ik}\sin\varphi_{qg} + \gamma_{qg}\right). \tag{16}$$

3.3. Containment of Clusters into a Container. Let us derive a phi-function representing analytically the containment constraint (3), $\widehat{\Lambda}_q(z_q) \subset \Omega \iff \operatorname{int}\widehat{\Lambda}_q(z_q) \cap \Omega^* = \varnothing$ for each $q \in I_N$, where $\Omega^* = R^2 \setminus \operatorname{int}\Omega$.

Let $\Phi^{T_i\Omega^*}(u_i)$ be a phi-function of objects $T_i(u_i)$ and Ω^*, $i \in \Xi_q$.

Proposition 4. *A continuous and everywhere defined function*

$$\Phi^{\widehat{\Lambda}_q \Omega^*}\left(z_q\right) = \min\left\{\Phi^{T_i\Omega^*}\left(u_i\right), i \in \Xi_q\right\} \tag{17}$$

is a phi-function of $\widehat{\Lambda}_q(z_q)$ and the object Ω^*.

Proof. We show that if $\widehat{\Lambda}_q(z_q) \subset \Omega$ then $T_i \subset \Omega \iff \operatorname{int} T_i(u_i) \cap \Omega^* = \varnothing, i \in \Xi_q$.

We assume that $\Omega^* = \Omega^*_1 \cup \Omega^*_2 \cup \Omega^*_3 \cup \Omega^*_4$, where $\Omega^*_s = \{(x, y) \mid g_s(x, y) \leq 0\}$ is a half plane, $s = 1, 2, 3, 4$, $g_1(x, y) = x_i$, $g_2(x, y) = y$, $g_3(x, y) = -x + l$, and $g_4(x, y) = -y + w$,

For each half plane Ω^*_s we can conclude that

(1) $\widehat{\Lambda}_q(z_q) \cap \Omega^*_s = \varnothing \iff \Lambda_q(z_q) \cap \Omega^*_s = \varnothing$ and therefore $T_i(u_i) \cap \Omega^*_s = \varnothing$, for all $i \in \Xi_q$;

(2) $\operatorname{int}\widehat{\Lambda}_q \cap \operatorname{int}\Omega^*_s \neq \varnothing \iff \operatorname{int}\Lambda_q(z_q) \cap \operatorname{int}\Omega^*_s \neq \varnothing$ and therefore there exists at least one object $T_i(u_i)$, such that $\operatorname{int} T_i(u_i) \cap \operatorname{int}\Omega^*_s \neq \varnothing, i \in \Xi_q$;

(3) $(\operatorname{int}\widehat{\Lambda}_q(z_q) \cap \operatorname{int}\Omega^*_s = \varnothing) \wedge (fr\,\widehat{\Lambda}_q \cap fr\Omega^*_s \neq \varnothing) \iff (\operatorname{int}\Lambda_q(z_q) \cap \operatorname{int}\Omega^*_s = \varnothing) \wedge (fr\,\Lambda_q(z_q) \cap fr\Omega^*_s \neq \varnothing)$ and therefore $\operatorname{int} T_i(u_i) \cap \operatorname{int}\Omega^*_s = \varnothing$, for all $i \in \Xi_q$, and there exists at least one object $T_i(u_i)$, such that $fr\,T_i(u_i) \cap fr\Omega^*_s \neq \varnothing$, $i \in \Xi_q$.

Thus, $\widehat{\Lambda}_q(z_q) \subset \Omega \iff \operatorname{int} T_i(u_i) \cap \Omega^* = \varnothing$, $i \in \Xi_q$. It means that $\Phi^{\widehat{\Lambda}_q\Omega^*}(z_q) \geq 0$ if and only if $\Phi^{T_i\Omega^*}(u_i) \geq 0$ for each $i \in \Xi_q$.

Now we state by phi-functions containment conditions for the object Ω^* and the following types of objects $T_i(u_i)$, $i \in \Xi_q$: an ellipse $E_i(u_i)$; a circle $C_i(v_i)$; and a convex polygon $K_i(u_i)$.

Phi-Function for an Ellipse Containment. A phi-function for an ellipse $E(u_i)$ given by its semiaxes (a_i, b_i) and the object Ω^* can be defined in the form

$$\Phi^{E_i\Omega^*}\left(u_i\right) = \min_{s=1,\ldots,4} g_{is}\left(u_i\right),$$

$$g_{i1}\left(u_i\right) = x_i - \sqrt{b_i^2 + \left(a_i^2 - b_i^2\right)\cos^2\theta_i},$$

$$g_{i2}\left(u_i\right) = y_i - \sqrt{b_i^2 + \left(a_i^2 - b_i^2\right)\sin^2\theta_i},$$

$$g_{i3}(u_i) = l - x_i - \sqrt{b_i^2 + (a_i^2 - b_i^2)\cos^2\theta_i},$$
$$g_{i4}(u_i) = w - y_i - \sqrt{b_i^2 + (a_i^2 - b_i^2)\sin^2\theta_i}. \quad (18)$$

The inequality $\Phi^{E_i\Omega^*}(u_i) \geq 0$ guarantees that $E_i(u_i) \subset \Omega$. We define the phi-function based on the idea of nonoverlapping of an ellipse and half plane introduced in [5].

Phi-Function for a Circle Containment. A phi-function for a circle $C(u_i)$ given by its radius r_i and the object Ω^* has the form

$$\Phi^{C_i\Omega^*}(u_i) = \min_{s=1,\dots,4} f_{is}(u_i),$$
$$g_{i1}(u_i) = x_i - r_i,$$
$$g_{i2}(u_i) = y_i - r_i, \quad (19)$$
$$g_{i3}(u_i) = -x_i + l - r_i,$$
$$g_{i4}(u_i) = -y_i + w - r_i,$$

The inequality $\Phi^{C_i\Omega^*}(u_i) \geq 0$ guarantees that $C_i(v_i) \subset \Omega$.

Phi-Function for a Convex Polygon Containment. A phi-function for a convex polygon $K(u_i)$ given by its vertices $p_{ik} = (p_{ik}^x, p_{ik}^y)$, $k = 1, \dots, l_i$, and the object Ω^* has the form

$$\Phi^{K_i\Omega^*}(u_i) = \min_{\substack{k=1,\dots,l_i \\ s=1,\dots,4}} g_s(p_{ik}^x, p_{ik}^y). \quad (20)$$

The inequality $\Phi^{K_i\Omega^*}(u_i) \geq 0$ guarantees that $K_i(u_i) \subset \Omega$. □

4. Mathematical Model

A mathematical model for layout clusters of objects in a rectangular domain Ω can be formulated in the form

$$\max_{(u,\phi,\psi,\rho)\in W\subset R^\sigma} \rho, \quad (21)$$

$$W = \Big\{(u,\phi,\psi,\rho) \in R^\sigma : \Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) - 0.5\rho \geq 0, q > g \in I_N, \rho \geq 0, \Phi'^{T_iT_j}(u_i, u_j, u_{ij}) \geq 0, i > j, (i,j) \in \Xi_q \times \Xi_q, q \in I_N, \Phi^{T_i\Omega^*}(u_i) \geq 0, i \in I_n\Big\}, \quad (22)$$

where ρ is considered as a distance between convex hulls $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$, $u = (u_1, u_2, \dots, u_n)$ is a vector of placement parameters, $u_i = (v_i, \theta_i)$, $v_i = (x_i, y_i)$, $\psi = (\varphi_{qg}, \gamma_{qg}, q > g \in J_N)$ is a vector of auxiliary variables in a quasi-phi-function of convex hulls $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$, $\phi = (u_{ij}, (i,j) \in \Xi_q \times \Xi_q, q \in I_N)$ is vector of auxiliary variables in a quasi-phi-function of objects $T_i(u_i)$ and $T_j(u_j)$, $\Phi'^{T_iT_j}(u_i, u_j, u_{ij})$ is a quasi-phi-function of objects $T_i(u_i)$ and $T_j(u_j)$, $\Phi^{T_i\Omega^*}(u_i)$ is a phi-function of objects $T_i(u_i)$ and Ω^* defined in (17), $\Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$ is a quasi-phi-function of convex hulls $\widehat{\Lambda}_q(z_q)$ and $\widehat{\Lambda}_g(z_g)$ defined in (10), and $\sigma = 1 + 3n + (\tau_q/2)\sum_{q=1}^N m_q(m_q - 1) + N(N-1)$ is the number of the problem variables. τ_q is a number of auxiliary variables for a function that describes nonoverlapping constraints for a pair of objects $T_i(u_i)$ and $T_j(u_j)$, $(i,j) \in \Xi_q \times \Xi_q$, $q \in I_N$. In particular, $\tau_q = 0$ for two circles, $\tau_q = 1$ for two ellipses, and $\tau_q = 2$ for two convex polygons (see Section 3.1).

Remark 5. In the case of two circles $C_i(u_i)$ and $C_j(u_j)$ we use a phi-function $\Phi^{C_iC_j}(u_i, u_j)$ instead of a quasi-phi-function $\Phi'^{T_iT_j}(u_i, u_j, u_{ij})$.

The feasible region W given by (22) is defined by a system of inequalities with differentiable functions. Our models (21)-(22) are a nonconvex and continuous nonlinear programming problem. This is an exact formulation in the sense that it gives all optimal solutions to the layout problem.

The models (21)-(22) involve $O(\sum_{q=1}^N m_q^2) + O(N^2)$ nonlinear inequalities and $O(\sum_{q=1}^N m_q^2) + O(N^2)$ variables due to the auxiliary variables in quasi-phi-functions.

We develop an efficient approach that employs a new algorithm for generating initial feasible solutions to search for local extrema of problems (21)-(22).

5. Solution Algorithm

Our solution strategy is based on the multistart algorithm and consists of three major stages: (1) generate a set of feasible starting points for the problems (21)-(22); (2) search for a set of local maxima for the problems (21)-(22) starting from each feasible point obtained at Stage (1); (3) choose the best local maxima from those found at Stage (2).

To generate feasible starting points of the problems (21)-(22) we develop a special algorithm **CSPA** (**C**luster **S**tarting **P**oints **A**lgorithm) that involves the following **Steps**.

Step 1. With each cluster $\Lambda_q(z_q)$ we associate a circular region λC_q with variable center point $v_q = (x_q, y_q)$ and variable radius λR_q, where λ is a scaling parameter and $R_q = \sqrt{\sum_{i\in\Xi_q} S_i}$, S_i is the area of object T_i, $i \in \Xi_q$, $q \in I_N$.

Step 2. Then we grow up all circular regions λC_q, $q \in I_N$, within a rectangular container Ω as much as possible, solving the following NLP subproblem:

$$\max_{(v,\lambda)\in V} \lambda, \quad (23)$$

$$V = \Big\{(v,\lambda) \in R^{2N+1} : \Phi^{C_qC_g}(v_q, v_g, \lambda) \geq 0, q > g \in I_N, \Phi^{C_q\Omega^*}(v_q, \lambda) \geq 0, q \in I_N, \lambda \geq 0\Big\}, \quad (24)$$

where $v = (v_1, v_2, \dots, v_N)$ is a vector of variable placement parameters of circular areas λC_q, $q \in I_N$;

$\Phi^{C_qC_g}(v_q, v_g, \lambda)$ is a phi-function of circular areas λC_q and λC_g with center points $v_q = (x_q, y_q)$ and $v_g = (x_g, y_g)$ and appropriate radii λR_q and λR_g, $q > g \in I_N$,

$$\Phi^{C_qC_g}\left(v_q, v_g, \lambda\right) = \left(x_q - x_g\right)^2 + \left(y_q - y_g\right)^2 - \left(\lambda R_q + \lambda R_g\right)^2; \tag{25}$$

$\Phi^{C_q\Omega^*}(v_q, \lambda)$ is a phi-function of a circular area λC_q and the object $\Omega^* = R^2 \setminus \operatorname{int}\Omega$, $q \in I_N$,

$$\begin{aligned}\Phi^{C_q\Omega^*}\left(v_q, \lambda\right) &= \min_{s=1,\dots,4} f_{qs}\left(v_q, \lambda\right),\\ f_{q1}\left(v_q, \lambda\right) &= x_q - \lambda R_q,\\ f_{q2}\left(v_q, \lambda\right) &= y_q - \lambda R_q,\\ f_{q3}\left(v_q, \lambda\right) &= l - x_q - \lambda R_q,\\ f_{q4}\left(v_q, \lambda\right) &= w - y_q - \lambda R_q.\end{aligned} \tag{26}$$

We search for a local maximum of problems (23)-(24) starting from a feasible point $(v^0 = (v_1^0, v_2^0, \dots, v_N^0), \lambda^0 = 0)$, where $v_q^0 \in \Omega$ and $q \in I_N$ are randomly generated points. Denote a local maximum point of problems (23)-(24) by (v^*, λ^*).

Step 3. Next we derive starting values of variables φ_{qg}^0, γ_{qg}^0 (obtained by trivial geometrical calculations); for each quasi-phi-function $\Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$, $q > g \in I_N$ in the problems (21)-(22). These variables are considered as parameters of separating lines between each pair of circular areas $\lambda^* C_q$ and $\lambda^* C_g$ of radii $\lambda^* R_q$ and $\lambda^* R_g$ with center points v_q^* and v_g^*, $q > g \in I_N$.

Step 4. Then we search for feasible points of the problems (21)-(22) solving the following NLP subproblem:

$$\max_{(u,\phi,\psi,\beta)\in G} \beta, \tag{27}$$

$$\begin{aligned}G = \Big\{&(u, \phi, \psi, \beta) \in R^\sigma : \Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}\left(z_q, z_g, \varphi_{qg}, \gamma_{qg}, \beta\right)\\ &\geq 0, q > g \in I_N, 0 \leq \beta \leq 1, \Phi'^{T_iT_j}\left(u_i, u_j, u_{ij}, \beta\right)\\ &\geq 0, i > j, (i, j) \in \Xi_q \times \Xi_q, q \in I_N, \Phi^{T_i\Omega^*}\left(u_i, \mu\right)\\ &\geq 0, i \in I_n\Big\},\end{aligned} \tag{28}$$

where $u = (u_1, u_2, \dots, u_n)$, $u_i = (v_i, \theta_i)$, $v_i = (x_i, y_i)$, and $\psi = (\varphi_{qg}, \gamma_{qg}, q > g \in J_N)$; β is a scaling parameter of our objects; $\phi = (u_{ij}, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$ is a vector of auxiliary variables in a quasi-phi-function of objects $\beta T_i(u_i)$ and $\beta T_j(u_j)$; $\Phi'^{T_iT_j}(u_i, u_j, u_{ij}, \beta)$ is a quasi-phi-function of objects $\beta T_i(u_i)$ and $\beta T_j(u_j)$; $\Phi^{T_i\Omega^*}(u_i)$ is a phi-function of objects $\beta T_i(u_i)$ and Ω^*;$\Phi'^{\widehat{\Lambda}_q\widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}, \beta)$ is a quasi-phi-function for two clusters defined in (10); $\sigma = 1 + 3n + (\tau_q/2)\sum_{q=1}^N m_q(m_q - 1) + N(N - 1)$ is the number of the problem variables, and τ_q is the number of auxiliary variables.

We use a feasible starting point $(u^0, \phi^0, \psi^0, \beta^0 = 0)$ to solve the problems (27)-(28), where $\psi^0 = (\varphi_{qg}^0, \gamma_{qg}^0, q > g \in J_N)$ is obtained at Step 3; $v_i^0 = (x_i^0, y_i^0) \in C_q$ is a vector of random generated translation parameters and $\theta_i^0 \in [0, 2\pi]$ is a randomly generated rotation parameter of object $\beta T_i(u_i)$ for $i \in \Xi_q, q \in I_N$; $\phi^0 = (u_{ij}^0, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$, and a vector u_{ij}^0 of auxiliary variables is found from the trivial geometrical calculations depending on the object shape (except circles) such that $\Phi'^{T_iT_j}(u_i^0, u_j^0, u_{ij}^0, \beta^0 = 0) \geq 0$. In particular, for two ellipses $u_{ij}^0 = \phi_{ij}^0$ and for two convex polygons $u_{ij}^0 = (\phi_{ij}^0, \gamma_{ij}^0)$ (see Section 3.1).

Global maximum of the problems (27)-(28) (i.e., $\beta^* = 1$) provides a feasible solution of the problems (21)-(22). A point of global maximum can be used as a starting point to search for local maximum of the problems (21)-(22).

Remark 6. Application of the decomposition algorithm described in [3] is recommended to solve the problem (21)-(22), when $m_q \geq 11$ for clusters $\Lambda_q(z_q)$, $q \in J_N$, in order to reduce the computational costs (time and memory).

6. Computational Results

Here we present a number of examples to demonstrate the efficiency of our methodology. We have run all experiments on an AMD FX(tm)-6100, 3.30 GHz computer, Programming Language C++, Windows 7. For the local optimisation we use the IPOPT code https://projects.coin-or.org/Ipopt), developed in [27]. Default options were used for running this software.

We present our new instances for the layout problem in a rectangular container $\Omega = \{(x, y) \in R^2 : 0 \leq x \leq l = 15, 0 \leq y \leq w = 15\}$. We run our program 100 times for each example.

Example 1. An ordered collection of $n = 30$ ellipses is given. The collection of ellipses is divided into $N = 3$ clusters $\Lambda_q(z_q)$, $q = 1, 2, 3 : m_1 = 10$, $m_2 = 10$, and $m_3 = 10$.

All ellipses are defined by their sizes $\{(a_i, b_i), i = 1, \dots, 30\}$ = { (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75), (0.9 0.6), (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75), (0.9 0.6), (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75) }.

Our Result

Placement Parameters of Ellipses. $\{(x_i, y_i, \theta_i), i = 1, \dots, 30\}$ = { (10.469949 12.267537 2.244218), (7.279733 12.352075 2.063582), (2.471444 12.548774 13.249926), (8.464111 9.691723 0.145847), (12.920038 11.433172 5.530417), (4.909339 12.890715 1.640578), (4.716335 10.894489 0.195532), (13.092110 13.335890 6.393843), (10.806375 9.993924 -0.401328), (6.267221 10.149879 6.032813), (0.813462 5.582920

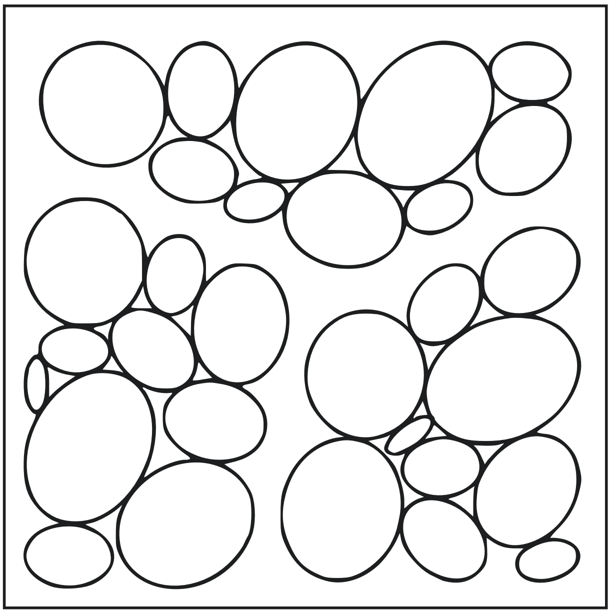

Figure 1: Local optimal arrangement of clusters in Example 1.

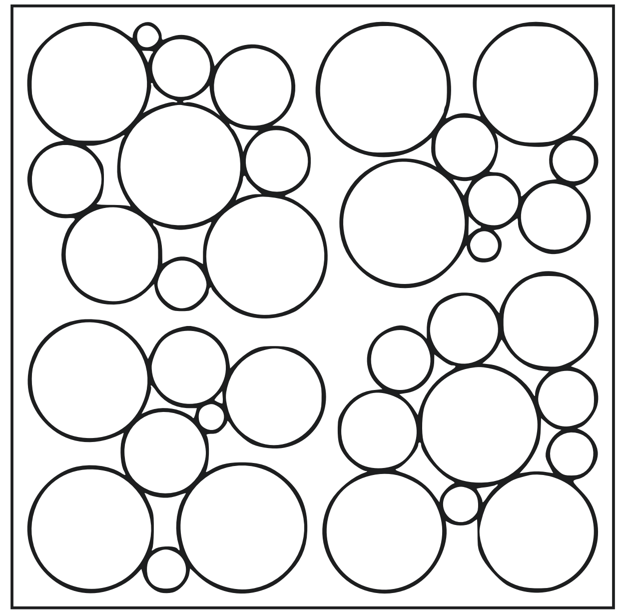

Figure 2: Local optimal arrangement of clusters in Example 2.

1.582319), (2.129866 4.016161 2.043699), (4.516627 2.106510 5.713416), (2.014535 8.617626 4.606157), (5.875381 7.080877 1.710554), (5.246435 4.663067 0.181324), (3.712908 6.439680 3.845887), (1.613133 1.313706 6.313737), (4.250763 8.301450 4.942235), (1.768064 6.435263 0.065954), (13.527926 1.206001 2.906999), (10.087641 4.308896 2.501728), (12.390656 5.699737 2.716022), (8.409176 2.472279 4.931800), (9.003824 5.818192 1.638294), (13.004545 2.937130 5.336177), (13.095024 8.392657 -0.572537), (10.944860 1.726727 0.720864), (10.969680 7.572605 5.373848), (10.898378 3.513018 -0.101262) }.

The value of objective function is $\rho^* = 2.109291$, while the computational time is 7131.446 sec.

Corresponding optimized layout is presented in Figure 1.

Example 2. An ordered collection of $n = 36$ circles is given. The collection of circles is divided into $N = 4$ clusters $\Lambda_q(z_q)$, $q = 1, 2, 3, 4 : m_1 = 10, m_2 = 10, m_3 = 8$, and $m_4 = 8$.

All circles are defined by their radii:

$\{(r_i), i = 1, \ldots, 36\}$ = { (1.495), (1.500), (1.501), (1.206), (1.009), (0.912), (0.815), (0.765), (0.621), (0.524), (0.336), (1.550), (1.533), (1.512), (1.239), (1.042), (0.945), (0.848), (0.7925), (0.654), (0.557), (0.380), (1.605), (1.566), (1.523), (1.272), (1.075), (0.978), (0.881), (0.820), (0.687), (0.590), (0.424), (1.660), (1.599), (1.534) }

Our Result

Placement Parameters of Circles. $\{(x_i, y_i), i = 1, \ldots, 36\}$= { (13.089089 1.910911), (9.282866 1.915911), (11.652953 4.566297), (13.378089 7.152616), (9.147148 4.421237), (11.269926 6.948703), (9.688888 6.189878), (13.819089 5.231585), (13.954597 3.846912), (11.189529 2.595038), (3.381819 14.237068), (4.209528 11.030014), (6.318116 8.780847), (1.927911 13.072089), (2.506044 8.821697), (6.003824 12.979437), (1.360911 10.681408), (6.604792 11.144527), (4.222026 13.460604), (4.247002 8.078382), (3.849810 0.972911), (4.971098 4.766312), (5.740826 2.020911), (1.981911 1.981911), (1.938911 5.681867), (6.544523 5.269738), (3.814234 3.883877), (4.418702 6.006885), (13.509939 9.758354), (11.287589 11.489670), (11.993830 10.158402), (13.994089 11.147396), (11.771968 9.049173), (9.264512 12.924089), (9.777002 9.600308), (13.050089 13.050089) }.

The value of objective function is $\rho^* = 0.415912$, while the computational time is 484.461sec.

Corresponding optimized layout is presented in Figure 2.

Example 3. An ordered collection of $n = 36$ objects is given. The collection of ellipses is divided into $N = 4$ clusters $\Lambda_q(z_q)$, $q = 1, 2, 3, 4 : m_1 = 10$ circles, $m_2 = 10$ circles, $m_3 = 8$ ellipses, and $m_4 = 8$ ellipses.

All circles are defined by their radii:

$\{(r_i), i = 1, \ldots, 20\}$={ (1.5), (1.5), (1.5), (1.2), (1.0), (0.9), (0.8), (0.75), (0.6), (0.5), (0.3), (1.5), (1.5), (1.5), (1.2), (1.0), (0.9), (0.8), (0.75), (0.6) }

All ellipses are defined by their semi axes:

$\{(a_i, b_i), i = 21, \ldots, 36\}$={ (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75), (0.9 0.6), (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5) }.

Our Result

Placement Parameters of Circles. $\{(x_i, y_i), i = 1, .., 20\}$={ (5.969333, 5.831559), (1.910221, 4.910224), (1.910221, 1.910223), (4.593503, 1.610221), (3.633171, 6.721701), (3.920715, 3.599532), (5.620213, 3.558210), (1.160222, 7.031545), (4.001869, 5.097335), (6.120123, 2.358173), (11.629726, 8.900163), (13.089779, 13.089779), (13.089779 10.089779), (8.950916, 11.115733), (10.406497,13.389779), (7.836995, 8.063426), (9.688425, 8.490281), (10.812773, 9.765367), (11.294872, 11.568861), (7.474932, 9.621922) }.

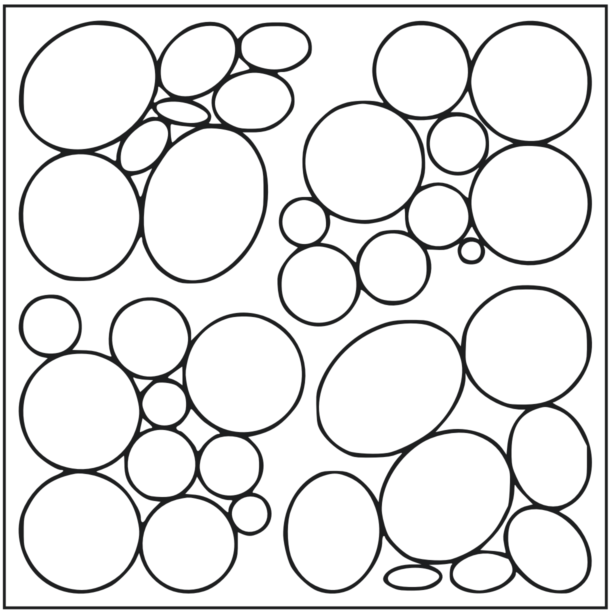

Figure 3: Local optimal arrangement of clusters in Example 3.

Placement Parameters of Ellipses. $\{(x_i, y_i, \theta_i), i = 21, .., 36\}$=(11.912729 0.916670 9.295412), (10.183793 0.775760 -0.039353), (9.620035 5.473407 -0.639827), (11.012582 2.782653 -3.922661), (12.998276 6.515842 -3.441852), (8.195878 1.909350 1.627631), (13.574899 3.775017 13.927174), (13.533974 1.475715 3.943306), (4.818922 13.635480 -0.705039), (6.200920 12.630733 3.035222), (6.736535 13.971098 -0.026565), (3.486483 11.514966 5.442422), (4.430982 12.359577 0.186450), (4.973013 10.072013 -1.233481), (2.113962 12.981281 2.518350), (1.910433 9.773977 17.233423) }.

The value of objective function is $\rho^* = 0.410222$, while the computational time is 1344.9 sec.

Corresponding optimized layout is presented in Figure 3.

Example 4. An ordered collection of $n = 36$ objects is given. The collection of objects is divided into $N = 4$ clusters $\Lambda_q(z_q)$, $q = 1, 2, 3, 4$; $m_1 = 10$ circles, $m_2 = 10$ circles, $m_3 = 8$ ellipses, and $m_4 = 8$ ellipses.

All circles are defined by their radii:

$\{(r_i), i = 1, \ldots, 20\}$={ (1.625), (1.575), (1.525), (1.275), (1.075), (0.975), (0.875), (0.8125), (0.675), (0.575), (0.400), (1.625), (1.575), (1.525), (1.275), (1.075), (0.975), (0.875), (0.8125), (0.675) }

All ellipses are defined by their semi axes:

$\{(a_i, b_i), i = 21, \ldots, 36\}$={ (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75), (0.9 0.6), (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5) }.

Our Result

Placement Parameters of Circles. $\{(x_i, y_i), i = 1, \ldots, 20\}$ = { (5.025933 1.876324), (1.826324 1.826324), (3.974219 6.518053), (6.563760 7.583078), (3.350845 3.993889), (1.451364 5.929808), (6.228908 4.078766), (1.109501 4.136463), (6.051897 5.654399), (4.881146 4.620133), (2.547234 11.199363), (1.876324 13.109992), (1.826324 9.360637), (7.011083 11.353004), (4.211329 11.390181), (8.183466 13.673676), (4.390190 13.773676), (6.234106 13.623755), (4.213121 9.302682), (5.668491 9.610177) }.

Placement Parameters of Ellipses. $\{(x_i, y_i, \theta_i), i = 21, \ldots, 36\}$ = { (11.954503 13.951223 14.239492), (11.755667 12.554007 2.556380), (9.866202 7.871985 -11.985510), (13.200316 7.946103 4.316218), (11.218788 10.594559 -25.265057), (13.543329 13.252969 -23.435600), (10.483036 13.231361 1.774510), (13.776518 10.616527 16.797305), (8.189442 2.594354 -21.480649), (7.771967 1.076809 22.539607), (14.052522 1.182705 20.790155), (9.223332 1.011216 -8.266075), (14.056688 2.541830 -19.537970), (11.441889 1.774003 -12.367135), (12.971454 4.412810 -15.416986), (9.596560 4.458588 44.002983) }.

The value of objective function is $\rho^* = 0.251325$, while the computational time is 1285.73 sec.

Corresponding optimized layout is presented in Figure 4.

Example 5. An ordered collection of $n = 36$ objects is given. The collection of objects is divided into $N = 4$ clusters $\Lambda_q(z_q)$, $q = 1, 2, 3, 4$: $m_1 = 10$ convex polygons, $m_2 = 10$ circles, $m_3 = 8$ ellipses, and $m_4 = 8$ ellipses.

All polygons are defined by their vertices:

$\{\{(p^x_{ik}, p^y_{ik}), k = 1, ..., 6\}, i = 1, \ldots, 10\}$ ={{ (1.975377 0.234652), (0.716736 1.400371), (-1.258641 1.165719), (-1.975377 -0.234652), (-0.716736 -1.400371), (1.258641 -1.165719) }, { (1.711902 0.463525), (0.374241 1.467221), (-1.337661 1.003696), (-1.711902 -0.463525), (-0.374241 -1.467221), (1.337661 -1.003696) }, { (1.425610 0.680986), (0.083738 1.497944), (-1.341873 0.816959), (-1.425610 -0.680986), (-0.083738 -1.497944), (1.341873 -0.816959) }, { (1.213525 0.705342), (-0.156793 1.193426), (-1.370318 0.488084), (-1.213525 -0.705342), (0.156793 -1.193426), (1.370318 -0.488084) }, {(0.919239 0.707107), (-0.336465 0.965926), (-1.255704 0.258819), (-0.919239 -0.707107), (0.336465 -0.965926), (1.255704 -0.258819) }, { (0.705342 0.728115), (-0.488084 0.822191), (-1.193426 0.094076), (-0.705342 -0.728115), (0.488084 -0.822191), (1.193426 -0.094076) }, { (0.499390 0.712805), (-0.599103 0.670936), (-1.098492 -0.041869), (-0.499390 -0.712805), (0.599103 -0.670936), (1.098492 0.041869) }, { (0.309017 0.713292), (-0.669131 0.557359), (-0.978148 -0.155934), (-0.309017 -0.713292), (0.669131 -0.557359), (0.978148 0.155934) }, { (0.140791 0.592613), (-0.699431 0.377592), (-0.840222 -0.215021), (-0.140791 -0.592613), (0.699431 -0.377592), (0.840222 0.215021) }, { (0.062767 0.498459), (-0.659301 0.283203), (-0.722068 -0.215256), (-0.062767 -0.498459), (0.659301 -0.283203), (0.722068 0.215256) }.

All circles are defined by their radii:

$\{r_i, i = 11, \ldots, 20\}$={ (0.336, 1.55, 1.533, 1.512, 1.239, 1.042, 0.945, 0.848, 0.7925, 0.654 }

All ellipses are defined by their semi axes:

$\{(a_i, b_i), i = 21, \ldots, 36\}$={ (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75), (0.9 0.6), (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5) }.

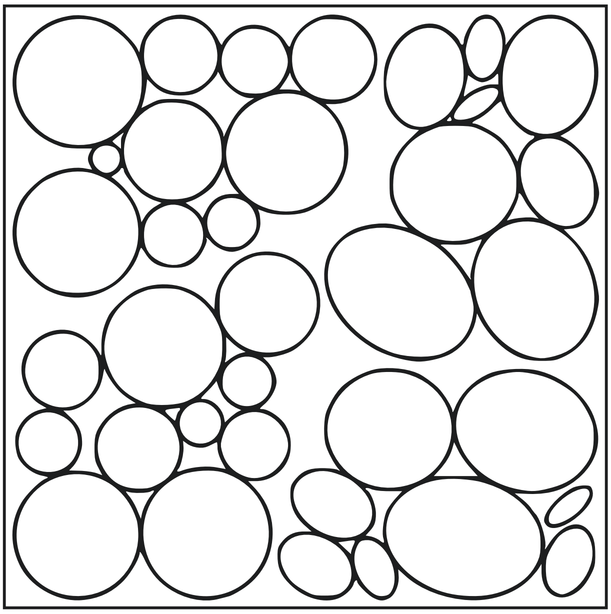

Figure 4: Local optimal arrangement of clusters in Example 4.

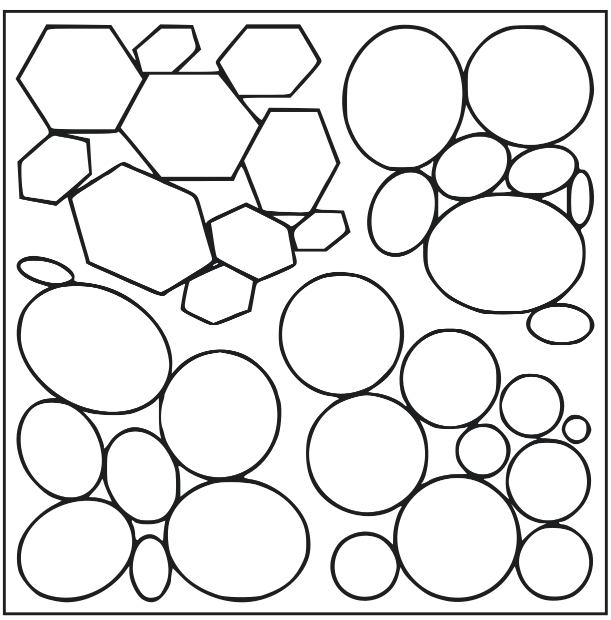

Figure 5: Local optimal arrangement of clusters in Example 5.

Our Result

Placement Parameters of Polygons. $\{(x_i, y_i, \theta_i), i = 1, \ldots, 10\}$ = { (3.447806 9.597127 2.606907), (4.632826 12.154203 6.031453), (1.935707 13.328641 2.695960), (7.137488 11.281372 4.553086), (6.594090 13.766083 3.344861), (6.187148 9.257646 -0.386618), (1.270813 11.108185 0.670004), (5.355596 7.966335 0.229233), (4.062287 14.055842 3.636614), (7.862803 9.559287 6.671758) }

Placement Parameters of Circles. $\{(x_i, y_i), i = 11, \ldots, 20\}$ = { (14.242145 4.607830), (11.278663 1.905800), (8.391616 6.969211), (9.039026 3.993831), (11.103865 5.862139), (13.546072 3.321421), (13.699200 1.300800), (8.985716 1.203801), (13.126975 5.194957), (11.917846 4.076973) }.

Placement Parameters of Ellipses. $\{(x_i, y_i, \theta_i), i = 21, \ldots, 36\}$ = ((3.646227 1.155469 1.533947), (1.038618 8.531725 0.246199), (2.256484 6.610724 0.489860), (5.799650 1.865204 6.453077), (5.363840 4.958834 1.597596), (1.798943 1.623608 5.811350), (1.423676 4.102206 7.386076), (3.423022 3.445736 4.374109), (9.931251 9.996600 5.101874), (11.636070 11.136792 2.665959), (13.383614 11.035564 2.838929), (13.844824 7.219741 6.333814), (14.343409 10.348285 1.605276), (12.459639 8.974386 0.003111), (9.963431 12.848701 1.699000), (13.051879 13.13605 10.285045) }.

The value of objective function is $\rho^* = 0.355801$, while the computational time is 4625.15 sec.

Corresponding optimized layout is presented in Figure 5.

Example 6. An ordered collection of $n = 30$ objects is given. The collection of objects is divided into $N = 3$ clusters $\Lambda_q(z_q)$, $q = 1, 2, 3 : m_1 = 10$ convex polygons, $m_2 = 10$ circles, and $m_3 = 10$ ellipses.

All polygons are defined by their vertices:

$\{\{(p^x_{ik}, p^y_{ik}), k = 1, \ldots, 6\}, i = 1, \ldots, 10\}$ = {{ (2.000000 0.000000), (1.000000 1.732051), (-1.000000 1.732051), (-2.000000 0.000000), (-1.000000 -1.732051), (1.000000 -1.732051) }, { (1.800000 0.000000), (0.900000 1.558846), (-0.900000 1.558846), (-1.800000 0.000000), (-0.900000 -1.558846), (0.900000 -1.558846) }, { (1.700000 0.000000), (0.850000 1.472243), (-0.850000 1.472243), (-1.700000 0.000000), (-0.850000 -1.472243), (0.850000 -1.472243) }, { (1.600000 0.000000), (0.800000 1.385641), (-0.800000 1.385641), (-1.600000 0.000000), (-0.800000 -1.385641), (0.800000 -1.385641) }, { (1.400000 0.000000), (0.700000 1.212436), (-0.700000 1.212436), (-1.400000 0.000000), (-0.700000 -1.212436), (0.700000 -1.212436) }, { (1.300000 0.000000), (0.650000 1.125833), (-0.650000 1.125833), (-1.300000 0.000000), (-0.650000 -1.125833), (0.650000 -1.125833) }, { 1.200000 0.000000), (0.600000 1.039230), (-0.600000 1.039230), (-1.200000 0.000000), (-0.600000 -1.039230), (0.600000 -1.039230) }, { (1.100000 0.000000), (0.550000 0.952628), (-0.550000 0.952628), (-1.100000 0.000000), (-0.550000 -0.952628), (0.550000 -0.952628) }, { (1.000000 0.000000), (0.500000 0.866025), (-0.500000 0.866025), (-1.000000 0.000000), (-0.500000 -0.866025), (0.500000 -0.866025) }, { (0.900000 0.000000), (0.450000 0.779423), (-0.450000 0.779423), (-0.900000 0.000000), (-0.450000 -0.779423), (0.450000 -0.779423) }}.

All circles are defined by their radii:

$\{r_i, i = 11, \ldots, 20\}$= { 0.336, 1.55, 1.533, 1.524, 1.513, 1.242, 1.23, 1.048, 0.951, 0.854 }.

All ellipses are defined by their semi axes:

$\{(a_i, b_i), i = 21, \ldots, 30\}$={ (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.9 1.6), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8) }.

Our Result

Placement Parameters of Polygons. $\{(x_i, y_i, \theta_i), i = 1, \ldots, 10\}$ = { (13.006728 7.204303 2.567940), (7.917914 11.613054

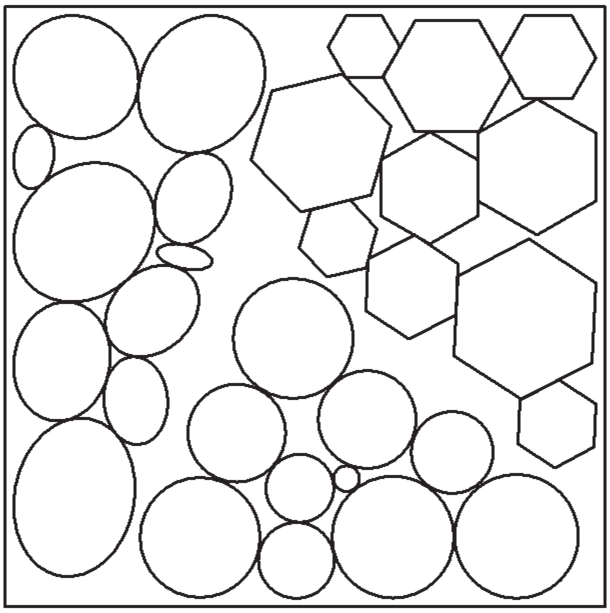

Figure 6: Local optimal arrangement of clusters in Example 6.

4.423250), (13.314400 11.008183 4.712389) (11.061231 13.271814 2.094380), (10.629721 10.486177 3.665191), (10.185459 8.006747 3.615137), (13.586642 13.747413 3.141592), (13.807690 4.598105 2.567940), (8.341396 9.221119 3.376054) (8.985800 14.007219 6.283185) }

Placement Parameters of Circles. $\{(x_i, y_i), i = 11, \ldots, 20\}$ = { (8.537177 3.208783), (12.787093 1.765862), (9.700995 1.746357), (4.878381 1.737357), (7.206433 6.712881), (5.794319 4.347302), (9.055276 4.686597), (11.178919 3.862321), (7.286139 1.164357), (7.371932 2.967318) }.

Placement Parameters of Ellipses. $\{(x_i, y_i, \theta_i), i = 21, \ldots, 30\}$ = { (0.720674 11.248417 4.850299), (4.490215 8.752444 3.338436), (1.734800 2.740444 1.764414), (1.981734 9.367337 5.537979), (4.912270 13.064601 2.125433), (1.778636 13.250446 3.779380), (1.425379 6.137135 4.902743), (3.677896 7.396278 2.407474), (4.702453 10.211496 2.031648), (3.268506 5.159948 1.494571) }.

The value of objective function is $\rho^* = 0.213358$, while the computational time is 7986.627 sec.

Corresponding optimized layout is presented in Figure 6.

7. Conclusions

We consider a new packing (layout) problem for a number of clusters of convex objects. This research is motivated by a container-loading problem; however, we think that similar problems arise naturally in many other packing/cutting/clustering issues. The layout is considered for a rectangular container of a given size subject to nonoverlapping of (continuously translated and rotated) objects within a cluster. The objects are specified by their sizes and have the same shape for the appropriate cluster. Each cluster is represented by the convex hull of objects that form this cluster. It is assumed that two clusters do not overlap each other if so are their convex hulls and a cluster belongs to a rectangular container if the corresponding convex hull does. New tools of mathematical modeling of nonoverlapping and containment for clusters are provided. An extension of the phi-function technique to the cluster convex hulls is presented taking into account variability of their shapes and metrical characteristics. New quasi-phi-functions and phi-functions are introduced. These functions avoid constructing convex hull for each cluster. A novel mathematical model for optimized layout of clusters is considered and stated as a nonlinear continuous problem. A novel algorithm for searching for initial feasible solutions to accelerate a local optimization procedure is developed. Computational results are provided to demonstrate the efficiency of our approach. The computational results are presented for new instances of the layout problem for clusters involving ellipses, circles, and convex polygons.

Appendix

Let $A \subset R^2$ and $B \subset R^2$ be two objects. The position of object A is defined by a vector of *placement parameters* $u_A = (v_A, \theta_A)$, where $v_A = (x_A, y_A)$ is a translation vector and θ_A is a rotation angle. The object A, rotated by an angle θ_A and translated by a vector v_A, is denoted by $A(u_A)$.

Definition [see [23]]. A continuous and everywhere defined function $\Phi^{AB}(u_A, u_B)$ is called a *phi-function* for objects $A(u_A)$ and $B(u_B)$ if

$$\begin{aligned}&\Phi^{AB}(u_A, u_B) < 0, \quad \text{if } \operatorname{int} A(u_A) \cap \operatorname{int} B(u_B) \neq \varnothing;\\ &\Phi^{AB}(u_A, u_B) = 0,\\ &\text{if } \operatorname{int} A(u_A) \cap \operatorname{int} B(u_B) = \varnothing \text{ and } frA(u_A) \cap frB(u_B) \neq \varnothing;\\ &\Phi^{AB}(u_A, u_B) > 0, \quad \text{if } A(u_A) \cap B(u_B) = \varnothing.\end{aligned} \tag{A.1}$$

Phi-functions allow us to distinguish the following three cases: $A(u_A)$ and $B(u_B)$ are intersecting so that $A(u_A)$ and $B(u_B)$ have common interior points; $A(u_A)$ and $B(u_B)$ do not intersect; i.e., $A(u_A)$ and $B(u_B)$ do not have common points; $A(u_A)$ and $B(u_B)$ are tangent; i.e., $A(u_A)$ and $B(u_B)$ have only common frontier points.

The inequality $\Phi^{AB}(u_A, u_B) \geq 0$ describes the nonoverlapping constraint, i.e., $\operatorname{int} A(u_A) \cap \operatorname{int} B(u_B) = \varnothing$, and the inequality $\Phi^{AB^*}(u_A, u_B) \geq 0$ describes the containment constraint $A(u_A) \subset B(u_B)$, i.e., $\operatorname{int} A(u_A) \cap \operatorname{int} B^*(u_B) = \varnothing$, where $B^* = R^2 \setminus \operatorname{int} B$.

Definition [see [23]]. A phi-function $\widetilde{\Phi}^{AB}(u_A, u_B)$ is called a *normalized phi-function* of $A(u_A)$ and $B(u_B)$ if its values coincide with the Euclidian distance between the objects $A(u_A)$ and $B(u_B)$, provided that $\operatorname{int} A(u_A) \cap \operatorname{int} B(u_B) = \varnothing$.

Let ρ be the minimum allowable distance between objects A and B, then $\widetilde{\Phi}^{AB} \geq \rho \Longleftrightarrow \operatorname{dist}(A, B) \geq \rho$. Here, $\operatorname{dist}(A, B) = \min_{a \in A, b \in B} d(a, b)$ and $d(a, b)$ is the Euclidean distance between two points a and b in R^2.

Definition [see [3]]. A continuous and everywhere defined function $\Phi'^{AB}(u_A, u_B, u')$ is called a quasi-phi-function for two objects $A(u_A)$ and $B(u_B)$ if $\max_{u'} \Phi'^{AB}(u_A, u_B, u')$ is a phi-function for the objects.

Here u' is a vector of auxiliary continuous variables that belongs to Euclidean space.

Based on features of a quasi-phi-function the nonoverlapping constraint can be described in the form: if $\Phi'^{AB}(u_A, u_B, u') \geq 0$ for some u', then $\operatorname{int} A(u_A) \cap \operatorname{int} B(u_B) = \varnothing$.

Definition [see [3]]. Function $\widetilde{\Phi}'^{AB}(u_A, u_B, u')$ is called a normalized quasi-phi-function for objects $A(u_A)$ and $B(u_B)$, if function $\max_{u'} \widetilde{\Phi}'^{AB}(u_A, u_B, u')$ is a normalized phi-function for the objects.

Data Availability

The data [input data for each example: the number of clusters, sizes, and numbers of objects that form each cluster; output data for each example: the value of objective function, computational time, and placement parameters of objects] used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

Research of the third author was partially supported by CONACYT (Mexico) [167019,293403].

References

[1] B. Chazelle, H. Edelsbrunner, and L. J. Guibas, "The complexity of cutting complexes," *Discrete & Computational Geometry*, vol. 4, no. 2, pp. 139–181, 1989.

[2] I. Litvinchev, L. Infante, and L. Ozuna, "Approximate packing: Integer programming models, valid inequalities and nesting," in *Optimized Packings with Applications*, G. Fasano and J. D. Pinter, Eds., vol. 105 of *Springer Optimization and Its Applications*, pp. 187–205, Springer, 2015.

[3] Y. Stoyan, A. Pankratov, and T. Romanova, "Quasi-phi-functions and optimal packing of ellipses," *Journal of Global Optimization*, vol. 65, no. 2, pp. 283–307, 2016.

[4] A. Pankratov, T. Romanova, and I. Litvinchev, "Packing ellipses in an optimized rectangular container," *Wireless Networks*, pp. 1–11, 2018.

[5] J. Kallrath and S. Rebennack, "Cutting ellipses from area-minimizing rectangles," *Journal of Global Optimization*, vol. 59, no. 2, pp. 405–437, 2014.

[6] E. G. Birgin, R. D. Lobato, and J. M. Martínez, "Packing ellipsoids by nonlinear optimization," *Journal of Global Optimization*, vol. 65, no. 4, pp. 709–743, 2016.

[7] P. I. Stetsyuk, T. E. Romanova, and G. Scheithauer, "On the global minimum in a balanced circular packing problem," *Optimization Letters*, vol. 10, no. 6, pp. 1347–1360, 2016.

[8] Z. Fu, W. Huang, and Z. Lü, "Iterated tabu search for the circular open dimension problem," *European Journal of Operational Research*, vol. 225, no. 2, pp. 236–243, 2013.

[9] Y. Stoyan and G. Yaskov, "Packing equal circles into a circle with circular prohibited areas," *International Journal of Computer Mathematics*, vol. 89, no. 10, pp. 1355–1369, 2012.

[10] H. Akeb, M. Hifi, and S. Negre, "An augmented beam search-based algorithm for the circular open dimension problem," *Computers & Industrial Engineering*, vol. 61, no. 2, pp. 373–381, 2011.

[11] I. Litvinchev, L. Infante, and L. Ozuna, "Packing circular-like objects in a rectangular container," *Journal of Computer and Systems Sciences International*, vol. 54, no. 2, pp. 259–267, 2015.

[12] J. Peralta, M. Andretta, and J. F. Oliveira, "Packing Circles and Irregular Polygons using Separation Lines," in *Proceedings of the International Conference on Operations Research and Enterprise Systems (ICORES)*, pp. 71–77, Funchal, Madeira, Portugal, 2018.

[13] J. Kallrath, "Cutting circles and polygons from area-minimizing rectangles," *Journal of Global Optimization*, vol. 43, no. 2-3, pp. 299–328, 2009.

[14] H. Alt and F. Hurtado, "Packing convex polygons into rectangular boxes," in *Proceedings of the Japanese Conference on Discrete and Computational Geometry (JCDCG)*, vol. 2098 of *Lecture Notes in Computer Science*, pp. 67–80, Springer, Tokyo, Japan, 2000.

[15] H. Alt, M. de Berg, and C. Knauer, "Approximating minimum-area rectangular and convex containers for packing convex polygons," *Journal of Computational Geometry*, vol. 8, no. 1, pp. 1–10, 2017.

[16] K. Tang, C. C. Wang, and D. Z. Chen, "Minimum area convex packing of two convex polygons," *International Journal of Computational Geometry & Applications*, vol. 16, no. 1, pp. 41–74, 2006.

[17] D. R. Jones, "A fully general, exact algorithm for nesting irregular shapes," *Journal of Global Optimization*, vol. 59, no. 2-3, pp. 367–404, 2014.

[18] F. M. B. Toledo, M. A. Carravilla, C. Ribeiro, J. F. Oliveira, and A. M. Gomes, "The dotted-board model: A new MIP model for nesting irregular shapes," *International Journal of Production Economics*, vol. 145, no. 2, pp. 478–487, 2013.

[19] Y. Stoyan, A. Pankratov, and T. Romanova, "Cutting and packing problems for irregular objects with continuous rotations: Mathematical modelling and non-linear optimization," *Journal of the Operational Research Society*, vol. 67, no. 5, pp. 786–800, 2016.

[20] Y. Stoyan, A. Pankratov, and T. Romanova, "Placement problems for irregular objects: Mathematical modeling, optimization and applications," in *Chapter in book Optimization Methods and Applications*, Butenko, Sergiy, Pardalos, M. Panos, Shylo, and Volodymyr, Eds., vol. 180 of *Springer Optimization and Its Applications Series*, p. 612, Springer, 2017.

[21] J. Bennell, G. Scheithauer, Y. Stoyan, T. Romanova, and A. Pankratov, "Optimal clustering of a pair of irregular objects," *Journal of Global Optimization*, vol. 61, no. 3, pp. 497–524, 2015.

[22] A. Bortfeldt and G. Wäscher, "Constraints in container loading—a state-of-the-art review," *European Journal of Operational Research*, vol. 229, no. 1, pp. 1–20, 2013.

[23] Y. Stoyan and T. Romanova, “Mathematical models of placement optimization: Two- and three-dimensional problems and applications,” in *Modeling and Optimization in Space Engineering*, G. Fasano and J. D. Pinter, Eds., vol. 73 of *Springer Optimization and Its Applications*, p. 404, Springer, 2012.

[24] N. Chernov, Yu. Stoyan, T. Romanova, and A. Pankratov, “Phi-functions for 2D objects formed by line segments and circular arcs,” *Advances in Operations Research*, vol. 2012, Article ID 346358, 26 pages, 2012.

[25] Y. Stoyan, T. Romanova, A. Pankratov, and A. Chugay, “Optimized object packings using quasi-phi-functions,” in *Optimized Packings and Their Applications*, G. Fasano and J. D. Pinter, Eds., vol. 105 of *Springer Optimization and Its Applications*, pp. 265–293, Springer, 2015.

[26] T. Romanova, J. Bennell, Y. Stoyan, and A. Pankratov, “Packing of concave polyhedra with continuous rotations using nonlinear optimisation,” *European Journal of Operational Research*, vol. 268, no. 1, pp. 37–53, 2018.

[27] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.