Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-12 (2016) 001–005

The 3D Object Packing Problem into a Parallelepiped Container Based on Discrete-Logical Representation

Mikhail Verkhoturov*. Alexander Petunin**
Galina Verkhoturova*, Konstantin Danilov*, Dmitry Kurennov**

**Ufa State Aviation Technical University, Ufa, Russia (e-mail: verhotur@vmk.ugatu.ac.ru).*
** *Ural Federal University, Yekaterinburg, Russia (e-mail: aapetunin@gmail.com)*

Abstract: The problem of 3D geometric objects irregular tight packing into minimal height cuboid is considered. Main approaches to solving this problem are described. The no-fit-polyhedron based algorithm using discrete-logical representation is proposed. Some examples and computational results are also given for public input data.

Keywords: packing, nesting, no-fit-polyhedron, non-intersection conditions of polyhedrons, 3-D packing problem, voxel.

1. INTRODUCTION

Analysis of complex products life cycle stages in different industries reveals that many of them require solving of the placement optimization tasks. Finding the optimal (or close) solutions can significantly reduce various resources consumption and production costs. Such problems are important in terms of saving resources, but are difficult to solve.

On the other hand, the emergence of additive technologies and rapid prototyping techniques revolutionized the high-tech industries, for instance aviation and aerospace industry, nuclear industry, medical and instrumentation. They are characterized as small-scale or piece production. Using new methods for the synthesis of forms and synthesis models by layering synthesis technology allowed to drastically reduce the time to create new products. Since a number of independent parts can be manufactured simultaneously, the implementation of such technologies leads to the necessity of solving the problem of the irregular 3D objects placement optimization, which is desirable from the standpoint of saving time, energy and other resources.

Many researchers worldwide are engaged in the study of cutting-packing problems. The most difficult one is complex-shaped 3D objects placement into given space (container) optimization. Analysis of published papers and review articles revealed that of 158 jobs during 1980-2011, only three investigated the problem of the irregular 3D object placement that is approximately 1.9%, Bortfeldt et al. (2013). The content of these articles, and other works that were not included in the above review leads to the conclusion that the study of ways to improve the effectiveness of the solutions (in terms of time spent and quality) is still relevant.

2. STATEMENT OF A PROBLEM

Suppose we have a set of 3D geometric objects (GO): $T = \{T_1, T_2, ..., T_n\}$: $T_i \subset \mathbf{R}^3, i = \overline{1,n}$, each in its own coordinates.

Layout area $Q \subset \mathbf{R}^3$ is a rectangular parallelepiped with variable height H, fixed length L and a width W.

Let $T_i(\bar{u}_i)$ is a geometric object T_i offset by vector $\bar{u}_i(x_i, y_i, z_i)$. Rotation is not considered in this paper.

Resulting positioning schema must fulfill the following conditions:

- Mutual non intersection:

$$T_i(\bar{u}_i) \cap T_j(\bar{u}_j) = \varnothing, \forall i = \overline{1,n}, \forall j = \overline{1,n}, i \neq j$$

- Being inside container

$$T_i(\bar{u}_i) \cap Q = T_i(\bar{u}_i), \forall i = \overline{1,n}$$

Equations (1) and (2) restrict possible placement parameters $U = (\bar{u}_1, \bar{u}_2, ..., \bar{u}_n) \in R^{3n}$ for objects set T inside area Q.

Let H = Z(Q(U)) to be minimal height to place all objects of $T = \{T_1, T_2, ..., T_n\}$ with offset vectors $U = \{\bar{u}_1, \bar{u}_2, ..., \bar{u}_n\}$.

Problem is to find a set of offset vectors U that minimize Z(T(U))->min, while restrictions (1) and (2) remains met (Fig. 1).

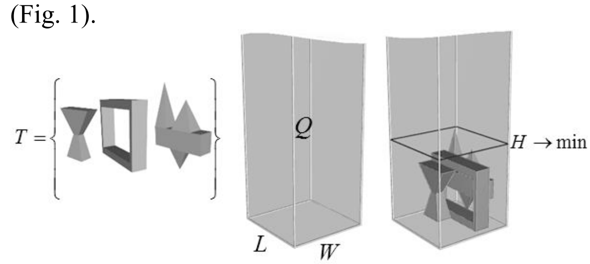


Fig. 1. Statement of 3D objects placement problem

In above terms, this problem is complex optimization of geometric modeling in high-dimensional space with nonconvex and disconnected space of possible solutions. It belongs to NP complexity class. In addition to

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.540

optimization, it has also geometric aspect to obey restrictions of mutual non-intersection and placement inside given layout space, Stoyan et al. (2009).

3. PROBLEM APPROACHES

Popular methods for solving 2D and 3D tasks of complex shaped geometric objects irregular placement are those of rational (permissible) pilings close to optimal. Usually they operate with single object at every single step of decision (object by object placement principle).
Solution process consists of the following procedures, named "encoding", "decoding" and "evaluating", Lutters (2012):

1. Optimization - ordering sequence of objects:
   - Generation of sequence of objects to place;
   - Reordering of objects;
2. Geometric procedure applied to objects according to their position in sequence:
   - Appropriate object representation (polygonal, voxel etc.);
   - Object motion modeling;
   - Choosing object position according to some criteria
   - Object placement into area with possible area growth

These procedures are often thus combined:

1. Generating object sequence (ordered list)
2. Sequence loop
   2.1. Object motion modeling
   2.2. Choice of object position according to some criteria
   2.3. Adding object to area (with possible area growth)
3. Calculating goal function

The loop is terminated after predefined iterations, time or when goal function reaches its limit.
A large variety of heuristics used for solving irregular placement problems at optimization phase exist. In most cases two methods classes are used. The first one is metaheuristics like "simulated annealing" (SA), "genetic algorithm" (GA), "tabu search" (TS), "ant colonies" (AC) with their modifications. The second one is heuristic methods crafted specifically for these problems.
In this study object sequence was built with "First match with ordering" algorithm, Garey et al. (1979). List is sorted according to object volumes in descending order.
Geometric procedures can be implemented in three ways:

1. Simulating object motion with mutual non-intersection (inside layout area), Heckmann et al. (1998)
2. Arbitrary motion (shifts and rotations), where object can overlap each other and layout area, Lutfiyya et al. (1991), Heckmann et al. (1995)
3. Positioning objects into arbitrary area, Blazewicz et al. (1993)

These methods differs in:

- Path of object movement
- Complexity of rotation modeling
- Whether object intersections are allowed during solution phases

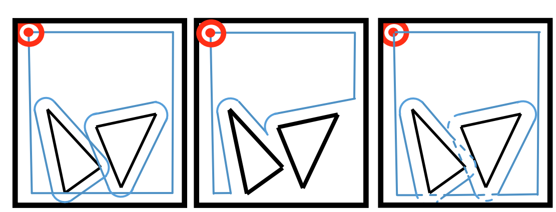
The one of the most wide used geometric methods is based upon modeling object movements inside layout area with restriction of their mutual non-intersection. It uses the concept of No-Fit-Polyhedron (NFP), Egeblad et al. (2007).
No-Fit-Polyhedron G_{12} or $G(T_1(0), T_2(u_2))$ for moving object $T_2(u_2)$ around fixed object T_1 is the set of T_2 positions where it is tightly fit to T_1.
NFP G_{12} of moving T_2 about fixed T_1 can be found using Minkowski operations, Pavlidis (1992):

$G_{12} = T_1(0) \oplus -(T_2(u_2))$, where

$A \oplus B = \{a + b | a \in A, b \in B\}$ - Minkowski sum of A and B sets

3.1. NFP USAGE SCENARIOS FOR OBJECT PLACING CONSIDERING ALREADY PLACED OBJECTS AND LAYOUT AREA

Several approaches for using NFP are known, Verhoturov (2012):
1. **Preliminary**. NFP for all object pairs and layout area are calculated beforehand (Fig. 2a). After object positioning, all NFP involved also shift according its new position.
2. **Integral**. For every object its NFP is calculated, as if already positioned objects were parts of layout area (Fig. 2b).
The main disadvantage of the first approach is that it assumes a lot of NFP calculation which will never be used.
The second approach often leads to unconnected layout area, that makes difficult to find available positions to place next object.
New “Dynamic” NFP scheme was developed to overcome these drawbacks. It allows avoiding excessive NFP calculations.
3. **Dynamic**. NFP for object to place is calculated for layout area and every placed object. Then every NFP is restricted using aforementioned package conditions (Fig. 2c).



a) b) c)

Fig. 2 NFP calculation scenarios (2D case).The circle is to be placed into rectangular area, where two triangles are already placed. a) Preliminary b) Integral c) Dynamic

NFP ALGORITHM WITH DYNAMIC SCHEME

Here follows the dynamic NFP scheme, Verhoturov (2012).
Suppose first (m-1) objects $\{T_1, T_2, \dots, T_n\}$ are already placed, having m-1<n. The next step is to position T_m object as follows: