Multidimensional Cube Packing

Yoshiharu Kohayakawa, Flavio Keidi Miyazawa, Prabhakar Raghavan, and Yoshiko Wakabayashi

Abstract. We consider the $d$-dimensional cube packing problem ($d$-CPP): given a list $L$ of $d$-dimensional cubes and (an unlimited quantity of) $d$-dimensional unit-capacity cubes, called bins, find a packing of $L$ into the minimum number of bins. We present two approximation algorithms for $d$-CPP, for fixed $d$. The first algorithm has an asymptotic performance bound that can be made arbitrarily close to $2 - \left(\frac{1}{2}\right)^d$. The second algorithm is an improvement of the first and has an asymptotic performance bound that can be made arbitrarily close to $2 - \left(\frac{2}{3}\right)^d$. To our knowledge, these results improve the bounds known so far for $d = 2$ and $d = 3$, and are the first results with bounds that are not exponential in the dimension.

Key Words. Approximation algorithms, Multidimensional bin packing, Asymptotic performance.

1. Introduction. We consider a generalization of the one-dimensional bin packing problem, called here the $d$-dimensional cube packing problem ($d$-CPP). Given a list $L$ of $n$ $d$-dimensional cubes (possibly of different sizes) and $d$-dimensional unit-capacity cubes, called bins, find an orthogonal packing of $L$ into the minimum number of bins. This problem is in fact a special case of the $d$-dimensional bin packing problem ($d$-BPP), in which one has to pack $d$-dimensional parallelepipeds into $d$-dimensional unit-capacity bins. Note that for $d = 1$ these problems coincide.

In 1989 Coppersmith and Raghavan [6] presented an online algorithm for $d$-BPP with asymptotic performance bound $(3 \cdot 2^d + 1)/4$. This algorithm, when specialized to $d$-CPP, has asymptotic performance bound $(\frac{3}{2})^d - (\frac{1}{2})^d + 1$. In 1999 Ferreira et al. [10] presented an asymptotic 1.99-approximation algorithm for 2-CPP; later, Miyazawa and Wakabayashi [16] presented a 2.67-approximation algorithm for 3-CPP. These algorithms can be generalized to an algorithm for $d$-CPP with asymptotic performance bound $(\frac{3}{2})^d - (\frac{2}{3})^d + 1$.

The most studied case is when $d = 1$. For this case there are asymptotic approximation schemes due to Karmarkar and Karp [11] and Fernandez de la Vega and Lueker [9]. For a recent survey on this case, see [5]. For the case $d = 2$, Chung et al. [3] obtained in the early eighties an asymptotic 2.125-approximation algorithm. Recently, Caprara [2] obtained a 1.691-approximation algorithm. For 3-BPP Li and Cheng [14] and Csirik and van Vliet [8] designed algorithms with the asymptotic performance bound 4.84. Their
algorithms generalize to \( d \)-BPP giving algorithms with the asymptotic performance bound close to \( 1.691^d \). For a survey on approximation algorithms for packing problems we refer to [4].

We present two approximation algorithms for \( d \)-CPP. The first algorithm has an asymptotic performance bound that can be made as close to \( 2 - \left( \frac{1}{2} \right)^d \) as desired. The second algorithm is an improvement of the first one and has an asymptotic performance bound that can be made as close to \( 2 - \left( \frac{2}{3} \right)^d \) as desired. For \( d = 2 \) and \( d = 3 \) the bounds are close to \( \frac{14}{9} \approx 1.56 \) and \( \frac{46}{27} \approx 1.70 \), respectively. To our knowledge, these results improve the bounds known so far for \( d = 2 \) and \( d = 3 \), and are the first results with bounds that are not exponential in the dimension (see the last paragraph of this section).

Recently, Seiden and van Stee [18] presented an algorithm for \( d = 2 \) with the bound \( \frac{14}{9} + \varepsilon \) that uses an idea similar to the one we present in this paper. It is most likely that the extended abstract [12], where we announced a comparable result, with explicit formulas for general \( d \) and a full description of our algorithms, was unknown to those authors.

The remainder of this paper is organized as follows. In the next section we present the notation and some definitions. In Section 3 we describe restricted versions of \( d \)-CPP. In Section 4 we present our approximation algorithms for \( d \)-CPP.

We remark that after the submission of this paper two new results have appeared, both improving the result we show in this paper: Bansal and Sviridenko [1] and Correa and Kenyon [7] have obtained (independently) an asymptotic polynomial-time approximation scheme for \( d \)-CPP.

2. Notation and Definitions. Given a cube \( c \), the size of \( c \), denoted by \( s(c) \), is defined as the length of an edge of \( c \). If \( L \) is a list of cubes, then we denote by \( V(L) \) the total volume of the cubes in \( L \). For a bin \( B \), we also denote by \( V(B) \) the volume of the cubes packed in \( B \), also called the volume occupancy of \( B \). Throughout this paper whenever we consider a list \( L \) to be packed into unit bins we suppose that all its cubes have size at most 1.

Given a list \( L \) of cubes, and an algorithm \( A \), we denote by \( A(L) \) the number of bins used by algorithm \( A \) when applied to \( L \), and by \( \text{OPT}(L) \) the number of bins used by an optimal packing of \( L \). If \( P \) is a packing of \( L \), we denote the number of bins used in \( P \) by \( |P| \). Some of our algorithms partition the input list \( L \) into sublists \( L_1, \ldots, L_k \) and then apply specialized algorithms for each sublist \( L_i \) generating a partial packing \( P_i \). We denote the packing obtained by the union of these packings by \( P_1 \cup \cdots \cup P_k \). We say that an algorithm \( A \) has an asymptotic performance bound \( \alpha \) if there exists a constant \( \beta \) such that \( A(L) \leq \alpha \cdot \text{OPT}(L) + \beta \) for all input lists \( L \). If \( \beta = 0 \), we also say that \( \alpha \) is an absolute performance bound for algorithm \( A \). We note that \( 2 \)-CPP cannot be approximated within \( 2 - \varepsilon \) in the absolute sense, unless \( P = \text{NP} \) (see [10], where this result is deduced from [13]). This and other negative results in terms of the absolute performance bounds make the asymptotic performance analyses of approximation algorithms for bin packing problems attractive.

3. Restricted \( d \)-CPP. Before considering the general problem, we present algorithms for restricted instances of \( d \)-CPP, to be used as subroutines.