

# An FPTAS for the Volume Computation of 0-1 Knapsack Polytopes Based on Approximate Convolution

Ei Ando<sup>1</sup> · Shuji Kijima<sup>2</sup>

Received: 31 January 2015 / Accepted: 9 November 2015  
© Springer Science+Business Media New York 2015

**Abstract** Computing high dimensional volumes is a hard problem, even for approximation. Several randomized approximation techniques for #P-hard problems have been developed in the three decades, while some deterministic approximation algorithms are recently developed only for a few #P-hard problems. Motivated by a new technique for a deterministic approximation, this paper is concerned with the *volume* computation of 0-1 *knapsack polytopes*, which is known to be #P-hard. This paper presents a new technique based on *approximate convolutions* for a *deterministic* approximation of volume computations, and provides a fully polynomial-time approximation scheme for the volume computation of 0-1 knapsack polytopes. We also give an extension of the result to multi-constrained knapsack polytopes with a constant number of constraints.

**Keywords** Approximate convolution · Volume computation · #P-hard · Knapsack polytope

## 1 Introduction

Computing a volume in the  $n$ -dimensional space is a hard problem, even for approximation. Lovász [17] showed that no polynomial-time *deterministic* algorithm can

---

A preliminary version appeared in [1].

---

✉ Ei Ando  
ando-ei@cis.sojo-u.ac.jp  
Shuji Kijima  
kijima@inf.kyushu-u.ac.jp

<sup>1</sup> Sojo University, Kumamoto, Japan

<sup>2</sup> Kyushu University, Fukuoka, Japan

approximate with ratio  $1.999^n$  the volumes of convex bodies in the  $n$  dimensional space as given by membership oracles (see also [3,9]).

Several *randomized* approximation techniques for #P-hard problems have been developed, such as the Markov chain Monte Carlo method. For the volume computation of a general convex body given by a membership oracle in the  $n$  dimensional space, Dyer et al. [8] gave the first *fully polynomial-time randomized approximation scheme* (FPRAS), giving a rapidly mixing Markov chain. In fact, the running time of the FPRAS is  $O^*(n^{23})$  where  $O^*$  ignores  $\text{poly}(\log n)$  and  $1/\epsilon$  terms. After several improvements, Lovász and Vempala [18] improved the time complexity to  $O^*(n^4)$ , and recently Cousins and Vempala [5] gave an  $O^*(n^3)$ -time algorithm.

In contrast, it is a major challenge to design *deterministic* approximation algorithms for #P-hard problems, and not many results seem to be known. A remarkable progress is the *correlation decay* argument due to Weitz [20]; he designed a *fully polynomial time approximation scheme* (FPTAS) for counting independent sets in graphs whose maximum degree is at least 5. A similar technique is independently presented by Bandyopadhyay and Gamarnik [2], and there are several recent developments on the technique, e.g., [4,10,13,14,16]. For counting 0-1 knapsack solutions, Gopalan et al. [11], and Stefankovic et al. [19] gave deterministic approximation algorithms based on the dynamic programming (see also [12]), in a similar way to a simple random sampling algorithm by Dyer [6]. Modifying the dynamic programming, Li and Shi [15] recently gave an FPTAS for computing a distribution function of sum of random variables, including the volume computation of 0-1 knapsack polytopes.

Motivated by a new technique of designing an FPTAS for #P-hard problems, this paper is concerned with the *volume* computation of 0-1 knapsack polytopes: Given a positive integer vector  $\mathbf{a}^\top = (a_1, \dots, a_n) \in \mathbb{Z}_{>0}^n$  and a positive integer  $b \in \mathbb{Z}_{>0}$ , the 0-1 knapsack polytope is given by

$$K \stackrel{\text{def}}{=} \{\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}^n \mid \mathbf{a}^\top \mathbf{x} \leq b, 0 \leq x_i \leq 1 (i = 1, \dots, n)\}.$$

Computing the *volume* of  $K$  is known to be #P-hard [7]. Remark that counting solutions corresponds to counting the integer points in  $K$ , and it is different from the volume computation, but closely related.

This paper presents a new technique for designing a *deterministic* approximation algorithm for volume computations, and provides an FPTAS for the volume computation of 0-1 knapsack polytopes, i.e., our main result is the following:

**Theorem 1.1** *For any  $\epsilon$  ( $0 < \epsilon \leq 1$ ), there exists an  $O(n^3/\epsilon)$ -time deterministic algorithm to approximate  $\text{Vol}(K)$  with approximation ratio  $1 + \epsilon$ .*

The idea of our algorithm is based on the classical *convolution*, while there seems no known technique (explicitly) using approximate convolutions to design an FPTAS for #P-hard problems. Our algorithm consists of recursively approximating convolutions with  $O(n^3/\epsilon)$  iterations, meaning that the number of iterations is independent of the values of  $a_i$  and  $b$ . In comparison with Li and Shi [15], their algorithm is based on the dynamic programming by Stefankovic et al. [19], and their FPTAS requires  $O((n^3/\epsilon^2) \log(1/\text{Vol}(K)) \log b)$  iterations, where notice that  $\text{Vol}(K) \leq 1$  holds by

the definition of  $K$ . Thus, the number of iterations in their algorithm depends on the input values  $\log a_i$  and  $\log b$ .

While a standard knapsack problem is characterized by a single nontrivial constraint, we also show in Sect. 4 that Theorem 1.1 is slightly extended to multi-constrained knapsack polytopes with a *constant* number of constraints, while a standard knapsack problem is characterized by a single nontrivial constraint. As far as we know, this is the first FPTAS for the problem. An extension of our technique to other polytopes is an interesting challenge.

This paper is organized as follows. In Sect. 2, we give a recursive convolution for the volume computation, to explain the idea of our algorithm. In Sect. 3, we explain our algorithm. In Sect. 4, we extend our technique to knapsack polytopes with a constant number of constraints. In Sect. 5, we give some supplemental proofs.

## 2 Preliminary: Convolution for $\text{Vol}(K)$

This section presents a convolution for  $\text{Vol}(K)$ , to explain the basic idea of our approximation algorithm. As a preliminary step, we “normalize” the problem for the convenience of arguments in this paper. Let  $M \in \mathbb{Z}_{>0}$  be an arbitrary constant for scaling<sup>1</sup>, and let

$$\tilde{a}_j \stackrel{\text{def}}{=} \frac{a_j}{b} M \quad (1)$$

for  $j = 1, 2, \dots, n$ . Let

$$\tilde{K} \stackrel{\text{def}}{=} \{\mathbf{x} \in [0, 1]^n \mid \tilde{\mathbf{a}}^\top \mathbf{x} \leq M\} \quad (2)$$

then, notice that

$$\begin{aligned} \tilde{K} &= \{\mathbf{x} \in [0, 1]^n \mid \tilde{\mathbf{a}}^\top \mathbf{x} \leq M\} \\ &= \{\mathbf{x} \in [0, 1]^n \mid \mathbf{a}^\top \mathbf{x} \leq b\} \\ &= K \end{aligned} \quad (3)$$

holds.

It may be helpful to introduce a probability space in order to describe a convolution of  $\text{Vol}(K)$ . Let  $\mathbf{X} = (X_1, \dots, X_n)$  be a uniform random variable over  $[0, 1]^n$ , i.e.,  $X_i$  ( $i = 1, 2, \dots, n$ ) are (mutually) independent. Then, it is not difficult to see that  $[\tilde{\mathbf{a}}^\top \mathbf{X} \leq M]$  if and only if  $[\mathbf{X} \in \tilde{K}]$ , meaning that

$$\Pr[\tilde{\mathbf{a}}^\top \mathbf{X} \leq M] = \Pr[\mathbf{X} \in \tilde{K}] = \frac{\text{Vol}(\tilde{K})}{\text{Vol}([0, 1]^n)} = \text{Vol}(\tilde{K})$$

<sup>1</sup> In fact,  $M$  is a scaling parameter of our approximation algorithm described in Sect. 3 in detail, that is  $M = 2\lceil n^2 \epsilon^{-1} \rceil$  for an approximate ratio  $\epsilon$  ( $0 < \epsilon \leq 1$ ). Arguments in this section also hold even in case that  $M$  is real. It might be convenient for this section to assume  $M = 1$ .

holds. Let  $f$  be the uniform density function over  $[0, 1]$ , i.e.,

$$f(x) = \begin{cases} 1 & 0 \leq x \leq 1, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

Now, we define  $\Phi_0(y) = 0$  for  $y \leq 0$ , while  $\Phi_0(y) = 1$  for  $y > 0$ . Inductively, we define for  $j = 1, 2, \dots, n$  that

$$\Phi_j(y) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} \Phi_{j-1}(y - \tilde{a}_j s) f(s) ds = \int_0^1 \Phi_{j-1}(y - \tilde{a}_j s) ds$$

for any  $y \in \mathbb{R}$ , where the last equality follows the definition (4) of  $f$ .

**Proposition 2.1** *For each  $j \in \{1, 2, \dots, n\}$  and for any  $y \in \mathbb{R}$ ,*

$$\Phi_j(y) = \Pr[\tilde{a}_1 X_1 + \dots + \tilde{a}_j X_j \leq y]$$

*holds.*

*Proof* Firstly, we prove the case of  $j = 1$ . Notice that  $\Phi_0$  is an indicator function in fact, then

$$\Phi_1(y) = \int_0^1 \Phi_0(y - \tilde{a}_1 s) ds = \Pr[y - \tilde{a}_1 X_1 \geq 0] = \Pr[\tilde{a}_1 X_1 \leq y]$$

and we obtain the claim in the case.

Next, inductively assuming that  $\Phi_{j-1}(y) = \Pr[\tilde{a}_1 X_1 + \dots + \tilde{a}_{j-1} X_{j-1} \leq y]$  holds, we prove the case of  $j$  ( $j = 1, 2, \dots, n$ ).

$$\begin{aligned} & \Pr[\tilde{a}_1 X_1 + \dots + \tilde{a}_j X_j \leq y] \\ &= \int_{-\infty}^{\infty} \Pr[\tilde{a}_1 X_1 + \dots + \tilde{a}_{j-1} X_{j-1} + \tilde{a}_j X_j \leq y \mid X_j = s] f(s) ds \\ &= \int_{-\infty}^{\infty} \Pr[\tilde{a}_1 X_1 + \dots + \tilde{a}_{j-1} X_{j-1} + \tilde{a}_j s \leq y] f(s) ds \\ &= \int_{-\infty}^{\infty} \Pr[\tilde{a}_1 X_1 + \dots + \tilde{a}_{j-1} X_{j-1} \leq y - s \tilde{a}_j] f(s) ds \\ &= \int_{-\infty}^{\infty} \Phi_{j-1}(y - \tilde{a}_j s) f(s) ds \\ &= \Phi_j(y). \end{aligned}$$

We obtain the claim. □

Let  $\Phi(M)$  denote  $\Phi_n(M)$ , for convenience. Then, Proposition 2.1, with the facts (3) and (4), suggests the following.

**Corollary 2.2**  $\Phi(M) = \text{Vol}(K)$ .

### 3 Approximation Algorithm for Vol( $K$ )

#### 3.1 The Idea

To begin with, this subsection explains the idea of our algorithm, that is a recursive computation of approximate convolutions  $G_j(y)$ , in an analogy with  $\Phi_j(y)$ . Let  $G_0(y) = \Phi_0(y)$ , i.e.,  $G_0(y) = 0$  for  $y \leq 0$  while  $G_0(y) = 1$  for  $y > 0$ . Inductively assuming  $G_{j-1}(y)$  ( $j = 1, \dots, n$ ), we define

$$\overline{G}_j(y) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} G_{j-1}(y - \tilde{a}_j s) f(s) ds = \int_0^1 G_{j-1}(y - \tilde{a}_j s) ds \quad (5)$$

for  $y \in \mathbb{R}$ , for convenience. Then, let  $G_j(y)$  be a staircase approximation of  $\overline{G}_j(y)$ , which is given by

$$G_j(y) \stackrel{\text{def}}{=} \overline{G}_j(\lceil y \rceil) \quad (6)$$

for  $y \in \mathbb{R}$ . We note the following fact, which may suggest a collateral evidence that  $\overline{G}_j$  approximates  $\Phi_j$ .

#### Proposition 3.1

$$\overline{G}_j(y) = \begin{cases} 0 & y \leq 0, \\ 1 & y \geq \sum_{j'=1}^j \tilde{a}_{j'}. \end{cases}$$

Notice that Proposition 2.1, in the previous section, suggests the same equation for  $\Phi_j(y)$ . See Sect. 5 for the proof of Proposition 3.1.

Here we briefly comment on the computation of  $G_j$ . Remark that it is sufficient to compute  $G_j(z)$  for  $z \in \mathbb{Z}$ , because  $G_j(y) = G_j(\lceil y \rceil)$  holds by the definition (6) of  $G_j(y)$ . Let

$$T \stackrel{\text{def}}{=} \{t \in \mathbb{Z}_{\geq 0} \mid t = z - s\tilde{a}_j \ (s \in [0, 1])\}, \quad (7)$$

i.e.,

$$T = \begin{cases} \{z, z-1, z-2, \dots, z - \lfloor \tilde{a}_j \rfloor\} & \text{if } z - \tilde{a}_j > 0, \\ \{z, z-1, z-2, \dots, 0\} & \text{otherwise.} \end{cases} \quad (8)$$

Let  $t_1, t_2, \dots, t_{|T|}$  denote the whole members in  $T$  such that  $t_k < t_{k+1}$  ( $k = 1, \dots, |T| - 1$ ). For convenience, let  $t_0 = z - \tilde{a}_j$  if  $z - \tilde{a}_j > 0$ , otherwise  $t_0 = 0$  ( $= t_1$ ). Let

$$\begin{aligned} d_s(u, v) &\stackrel{\text{def}}{=} |\{s \in \mathbb{R} \mid u \leq s\tilde{a}_j \leq v\}| \\ &= \frac{v - u}{\tilde{a}_j} \end{aligned} \quad (9)$$

for  $u, v \in \mathbb{R}$  such that  $u \leq v$ . Then,

$$\begin{aligned} G_j(z) &= \int_0^1 G_{j-1}(z - s\tilde{a}_j) ds \\ &= \sum_{k=0}^{|T|} d_s(t_{k-1}, t_k) G_{j-1}(t_k) \\ &= \begin{cases} \sum_{l=0}^{|T|-1} \frac{1}{\tilde{a}_j} G_{j-1}(z - l) + \frac{\tilde{a}_j - \lfloor \tilde{a}_j \rfloor}{\tilde{a}_j} G_{j-1}(z - \lfloor \tilde{a}_j \rfloor) & \text{if } z - \tilde{a}_j > 0, \\ \sum_{l=0}^{|T|-1} \frac{1}{\tilde{a}_j} G_{j-1}(z - l) & \text{otherwise.} \end{cases} \end{aligned} \quad (10)$$

Thus, we see that the value of  $G_j(z)$  is naively computed from the values  $G_{j-1}(z')$  ( $z' \in \{1, 2, \dots, z\}$ ) and  $\tilde{a}_j$  in  $O(z)$  time. We will later discuss a fast computation of  $G_j(z)$  based on a dynamic programming (see Lemma 3.3, and also Sect. 3.4 for detail).

### 3.2 Algorithm

Based on the arguments in Sect. 3.1, our algorithm is described as follows.

**Algorithm 1** Input<sup>2</sup>:  $\tilde{a} = (\tilde{a}_1, \dots, \tilde{a}_n) \in \mathbb{Q}_{>0}^n$ .

1. Let  $G_0(y) := 0$  for  $y \leq 0$ , and let  $G_0(y) := 1$  for  $y > 0$ ;
2. For  $j = 1, \dots, n$
3.   For  $z = 1, \dots, M$
4.     Compute  $G_j(z)$ ;
5. Output  $G_n(M)$ .

For convenience, let  $G(M)$  denote  $G_n(M)$ , in the following. The following proposition about the time complexity of Algorithm 1 is easy from the arguments in Sect. 3.1.

**Proposition 3.2** *The running time of Algorithm 1 is  $O(nM^2)$ .*

*Proof* As stated in Sect. 3.1,  $G_j(z)$  is computed at line 4 in  $O(M)$  time. Now, the claim is easy.  $\square$

In fact, the time complexity of lines 3 and 4 in Algorithm 1, that is  $O(M^2)$ , is improved to  $O(M)$  by a dynamic programming. We will establish the following lemma later in Sect. 3.4.

**Lemma 3.3** *The running time of Algorithm 1 is  $O(nM)$ .*

<sup>2</sup> For the simplicity of a description, here we assume that our input is given by rationals  $\tilde{a}_j = (a_j/b)M$  ( $j = 1, \dots, n$ ), since  $G_j$  is defined for  $\tilde{a}_j$  [see (5) and (6)]. It is also noted that  $M$  is a parameter of the algorithm (see Lemma 3.4, appearing later).

The approximation ratio of Algorithm 1 is given by the following lemma, which we will prove in the next subsection.

**Lemma 3.4** *For any  $\epsilon$  ( $0 < \epsilon \leq 1$ ), set an integer  $M \geq 2n^2\epsilon^{-1}$ , then we have*

$$\Phi(M) \leq G(M) \leq (1 + \epsilon)\Phi(M).$$

Theorem 1.1 is immediate from Lemmas 3.3 and 3.4.

### 3.3 Proof of Lemma 3.4, for Approximation Ratio

We prove Lemma 3.4. As a preliminary, we observe the following facts from the definitions of  $\overline{G}_j(y)$  and  $G_j(y)$ .

**Observation 3.5** *For any  $j \in \{1, 2, \dots, n\}$ ,  $\Phi_j(y)$ ,  $\overline{G}_j(y)$  and  $G_j(y)$  are monotone nondecreasing (with respect to  $y$ ), respectively.*

**Observation 3.6** *For each  $j \in \{1, 2, \dots, n\}$ ,  $\overline{G}_j(y) \leq G_j(y) \leq \overline{G}_j(y+1)$  holds for any  $y \in \mathbb{R}$ .*

**Proposition 3.7** *For each  $j \in \{1, 2, \dots, n\}$ ,  $\Phi_j(y) \leq \overline{G}_j(y)$  for any  $y \in \mathbb{R}$ .*

*Proof* By the definition,  $\Phi_0(y) = \overline{G}_0(y)$  for any  $y \in \mathbb{R}$ . Inductively assuming that  $\Phi_{j-1}(y) \leq \overline{G}_{j-1}(y)$  holds for any  $y \in \mathbb{R}$ , we have that

$$\begin{aligned} \Phi_j(y) &= \int_{-\infty}^{\infty} \Phi_{j-1}(y - \tilde{a}_j s) f(s) ds \\ &\leq \int_{-\infty}^{\infty} \overline{G}_{j-1}(y - \tilde{a}_j s) f(s) ds && \text{(Induction hypo.)} \\ &\leq \int_{-\infty}^{\infty} G_{j-1}(y - \tilde{a}_j s) f(s) ds = \overline{G}_j(y) && \text{(by Obs. 3.6)} \end{aligned}$$

for any  $y \in \mathbb{R}$ . □

Now, we establish the following Lemma 3.8, which claims a “horizontal” approximation ratio, in spite of Lemma 3.4 claiming a “vertical” bound.

**Lemma 3.8** *For each  $j \in \{1, \dots, n\}$ ,*

$$\Phi_j(y) \leq G_j(y) \leq \Phi_j(y + j)$$

*holds for any  $y \in \mathbb{R}$ .*

*Proof* The former inequality is immediate from Proposition 3.7 and Observation 3.6. Now, we are concerned with the latter inequality. Observation 3.6 implies that

$$G_1(y) \leq \overline{G}_1(y+1) = \Phi_1(y+1),$$

and we obtain the claim for  $j = 1$ . Inductively assuming that  $G_{j-1}(y) \leq \Phi_{j-1}(y + j - 1)$  holds for any  $y \in \mathbb{R}$ , we show that  $G_j(y) \leq \Phi_j(y + j)$ . Notice that

$$\begin{aligned} \overline{G}_j(y) &= \int_{-\infty}^{\infty} G_{j-1}(y - \tilde{a}_j s) f(s) ds \\ &\leq \int_{-\infty}^{\infty} \Phi_{j-1}(y - \tilde{a}_j s + j - 1) f(s) ds \quad (\text{induction hypo.}) \\ &= \Phi_j(y + (j - 1)) \end{aligned} \quad (11)$$

holds for any  $y \in \mathbb{R}$ . Using Observation 3.6, (11) implies that

$$G_j(y') \leq \overline{G}_j(y' + 1) \leq \Phi_j((y' + 1) + (j - 1)) = \Phi_j(y' + j)$$

and we obtain the claim.  $\square$

To transform the horizontal bound to a vertical one, we prepare the following.

### Lemma 3.9

$$\frac{\Phi(M)}{\Phi(M + n)} \geq \left( \frac{M}{M + n} \right)^n.$$

*Proof* For convenience, let  $\tilde{K}(y) \stackrel{\text{def}}{=} \{\mathbf{x} \in [0, 1]^n \mid \tilde{\mathbf{a}}^\top \mathbf{x} \leq y\}$  for  $y \geq 0$ , i.e.,  $\Phi(M) = \text{Vol}(\tilde{K}(M))$  and  $\Phi(M + n) = \text{Vol}(\tilde{K}(M + n))$ . Let  $H_M$  be a hyperplane defined by  $H_M \stackrel{\text{def}}{=} \{\mathbf{x} \in \mathbb{R}^n \mid \tilde{\mathbf{a}}^\top \mathbf{x} = M\}$ , and let  $F_M = \tilde{K}(M) \cap H_M$  i.e.,  $F_M = \{\mathbf{x} \in \tilde{K}(M) \mid \tilde{\mathbf{a}}^\top \mathbf{x} = M\}$ . Let  $C(y)$  for  $y \geq M$  be (a truncation of) a cone defined by

$$C(y) \stackrel{\text{def}}{=} \{\lambda \mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} \in F_M, \lambda \in \mathbb{R} \text{ such that } 0 \leq \lambda \leq y/M\}.$$

Notice that  $C(y) \cap H_M = F_M$  for any  $y \geq M$ . Figure 1 illustrates  $\tilde{K}(M)$ ,  $\tilde{K}(M + n)$ ,  $H_M$ ,  $F_M$ ,  $C(M)$ , and  $C(M + n)$ . It is not difficult from the definitions to see that  $C(M) \subseteq \tilde{K}(M)$  holds, since  $\tilde{K}(M)$  is convex and it contains  $F_M$  and the origin  $\mathbf{0}$ .

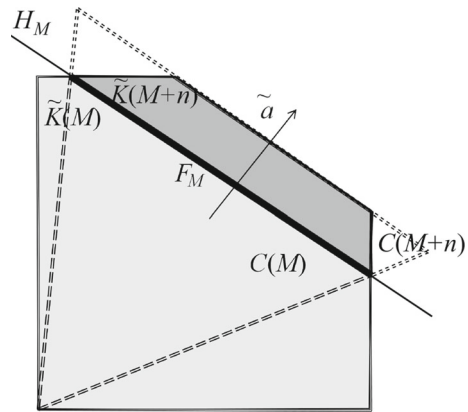
Next, we claim that  $C(M + n) \setminus C(M) \supseteq \tilde{K}(M + n) \setminus \tilde{K}(M)$  holds. Let  $\mathbf{u} \in \tilde{K}(M + n) \setminus \tilde{K}(M)$ . For convenience, let  $z = \tilde{\mathbf{a}}^\top \mathbf{u}$ , then  $M < z \leq M + n$  holds. Let  $\mathbf{u}' \stackrel{\text{def}}{=} (M/z)\mathbf{u}$  then  $\mathbf{u}' \in F_M$  since  $\tilde{\mathbf{a}}^\top \mathbf{u}' = (M/z)\tilde{\mathbf{a}}^\top \mathbf{u} = M$ . This implies that  $\mathbf{u} = (z/M)\mathbf{u}' \in C(M + n)$ .

Now, we obtain that

$$\begin{aligned} \Phi(M + n) &= \text{Vol}(\tilde{K}(M)) + \text{Vol}(\tilde{K}(M + n) \setminus \tilde{K}(M)) \\ &\leq \text{Vol}(\tilde{K}(M)) + \text{Vol}(C(M + n) \setminus C(M)) \end{aligned}$$



**Fig. 1** An illustration of  $\tilde{K}(M)$ ,  $\tilde{K}(M+n)$ ,  $H_M$ ,  $F_M$ ,  $C(M)$ , and  $C(M+n)$



where we remark that  $C(M+n) \setminus \tilde{K}(M) = C(M+n) \setminus C(M)$  for the last inequality. Then,

$$\begin{aligned} \frac{\Phi(M)}{\Phi(M+n)} &\geq \frac{\text{Vol}(\tilde{K}(M))}{\text{Vol}(\tilde{K}(M)) + \text{Vol}(C(M+n) \setminus C(M))} \\ &\geq \frac{\text{Vol}(C(M))}{\text{Vol}(C(M)) + \text{Vol}(C(M+n) \setminus C(M))} \quad (\text{since } \tilde{K}(M) \supseteq C(M)) \\ &= \frac{\text{Vol}(C(M))}{\text{Vol}(C(M+n))} \\ &= \left( \frac{M}{M+n} \right)^n \end{aligned}$$

where the last equality follows  $C(M)$  is similar to  $C(M+n)$  in the  $n$  dimensional space.  $\square$

Now, Lemma 3.4 is easy from Lemmas 3.8 and 3.9.

*Proof (of Lemma 3.4)* The first inequality  $\Phi(M) \leq G(M)$  is immediate from Lemma 3.8. Lemma 3.9 suggests that

$$\begin{aligned} \frac{\Phi(M)}{\Phi(M+n)} &\geq \left( \frac{M}{M+n} \right)^n = \left( \frac{1}{1 + \frac{n}{M}} \right)^n \\ &\geq \left( 1 - \frac{n}{M} \right)^n \quad \left( \text{since } \left( 1 + \frac{n}{M} \right)^n \left( 1 - \frac{n}{M} \right)^n < 1 \right) \\ &\geq \left( 1 - \frac{\epsilon}{2n} \right)^n \quad \left( \text{since } M \geq 2n^2\epsilon^{-1} \right) \\ &\geq 1 - n \frac{\epsilon}{2n} = 1 - \frac{\epsilon}{2} \end{aligned}$$

holds. Thus,

$$\frac{\Phi(M+n)}{\Phi(M)} \leq \frac{1}{1-\frac{\epsilon}{2}} \leq 1+\epsilon$$

for  $\epsilon \leq 1$ , and we obtain the claim.  $\square$

### 3.4 Proof of Lemma 3.3, for the Time Complexity

For a proof of Lemma 3.3, this subsection explains a dynamic programming to compute  $G_j(z)$  for  $z = 0, 1, \dots, M$  from  $G_{j-1}(z')$  ( $z' = 0, 1, \dots, M$ ).

Recall (8) [see also (7)], which is rewritten as

$$T = \begin{cases} \{z - \lfloor \tilde{a}_j \rfloor, z - \lfloor \tilde{a}_j \rfloor + 1, \dots, z\} & \text{if } z - \tilde{a}_j > 0, \\ \{0, 1, \dots, z\} & \text{otherwise,} \end{cases}$$

as well as recall (10), that is rewritten as

$$\begin{aligned} G_j(z) &= \sum_{k=0}^{|T|} d_s(t_{k-1}, t_k) G_{j-1}(t_k) \\ &= \begin{cases} \sum_{k=1}^{\lfloor \tilde{a}_j \rfloor} \frac{1}{\tilde{a}_j} G_{j-1}(k) + \frac{\tilde{a}_j - \lfloor \tilde{a}_j \rfloor}{\tilde{a}_j} G_{j-1}(z - \lfloor \tilde{a}_j \rfloor) & \text{if } z > \lfloor \tilde{a}_j \rfloor, \\ \sum_{k=0}^z \frac{1}{\tilde{a}_j} G_{j-1}(k) & \text{otherwise,} \end{cases} \end{aligned}$$

where we remark for the last equality that  $|T| = \lfloor \tilde{a}_j \rfloor$  holds in the case of  $z - \tilde{a}_j > 0$ , and that  $|T| = z$  in the other case.

Thus, it is not difficult to see that

$$G_j(z) = G_j(z-1) + \frac{1}{\tilde{a}_j} G_{j-1}(z) \quad (12)$$

holds when  $z \leq \lfloor \tilde{a}_j \rfloor$ . Similarly, in case that  $z = \lfloor \tilde{a}_j \rfloor + 1$ , we see that

$$G_j(z) = G_j(z-1) + \frac{1}{\tilde{a}_j} G_{j-1}(z) - \left( \frac{1}{\tilde{a}_j} - \frac{\tilde{a}_j - \lfloor \tilde{a}_j \rfloor}{\tilde{a}_j} \right) G_{j-1}(z - \lfloor \tilde{a}_j \rfloor) \quad (13)$$

holds, as well as in the other case, that is  $z > \lfloor \tilde{a}_j \rfloor + 1$ , we obtain that

$$\begin{aligned} G_j(z) &= G_j(z-1) + \frac{1}{\tilde{a}_j} G_{j-1}(z) - \left( \frac{1}{\tilde{a}_j} - \frac{\tilde{a}_j - \lfloor \tilde{a}_j \rfloor}{\tilde{a}_j} \right) G_{j-1}(z - \lfloor \tilde{a}_j \rfloor) \\ &\quad - \frac{\tilde{a}_j - \lfloor \tilde{a}_j \rfloor}{\tilde{a}_j} G_{j-1}(z - \lfloor \tilde{a}_j \rfloor - 1). \end{aligned} \quad (14)$$

Equations (12), (13) and (14) suggest that  $G_j(z)$  is computed from  $G_j(z-1)$  in  $O(1)$  time. It implies that the time complexity of computing  $G_j(z)$  for  $z = 0, 1, \dots, M$  is  $O(M)$ , in total. Now, Lemma 3.3 is easy. The whole procedure of computing  $G_j(z)$  for  $z = 0, 1, \dots, M$  from  $G_{j-1}(z')$  ( $z' = 0, 1, \dots, M$ ) is described as follows.

**Procedure 2** (Compute  $G_j$  from  $G_{j-1}$ , for lines 3–4 in Algorithm 1)

1. **Read**  $G_{j-1}(z)$  for  $z \in \{0, 1, \dots, M\}$ .
2. **Set**  $G_j(0) := 0$ ;
3. **For** ( $z = 1, 2, \dots, M$ )
4.   **If**  $z \leq \lfloor \tilde{a}_j \rfloor$ , **then Compute**  $G_j(z)$  from  $G_j(z-1)$  by (12);
5.   **Elseif**  $z = \lfloor \tilde{a}_j \rfloor + 1$ , **then Compute**  $G_j(z)$  from  $G_j(z-1)$  by (13);
6.   **Else** (i.e.,  $z > \lfloor \tilde{a}_j \rfloor + 1$ ), **Compute**  $G_j(z)$  from  $G_j(z-1)$  by (14);

## 4 Knapsack Polytope with a Constant Number of Constraints

Toward an extension of the technique to general polytopes, this section shows that the technique described in Sects. 2 and 3 is extended to multi-constrained knapsack polytopes with a constant number of constraints.

### 4.1 Problem Description and Preliminary

Suppose that  $m$  nonnegative integer vectors  $\mathbf{a}_i^\top = (a_{i1}, \dots, a_{in}) \in \mathbb{Z}_{\geq 0}^n$  for  $i = 1, \dots, m$ , and another positive integer vector  $\mathbf{b} = (b_1, \dots, b_m)^\top \in \mathbb{Z}_{>0}^m$  are given. For convenience, let  $A \in \mathbb{Z}_{\geq 0}^{m \times n}$  be a matrix defined by

$$A = \begin{pmatrix} \mathbf{a}_1^\top \\ \vdots \\ \mathbf{a}_m^\top \end{pmatrix} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,n} \\ & \ddots & \\ a_{m,1} & \cdots & a_{m,n} \end{pmatrix} = (A_1 \quad \cdots \quad A_n) \quad (15)$$

where  $A_j$  ( $j = 1, \dots, n$ ) denotes the column vectors of  $A$ , i.e.,  $A_j = (a_{1,j}, \dots, a_{m,j})^\top$ . Then, the knapsack polytope with  $m$  constraints  $K_m$  is given by

$$K_m \stackrel{\text{def}}{=} \left\{ \mathbf{x} \in [0, 1]^n \mid \bigwedge_{i=1, \dots, m} \mathbf{a}_i^\top \mathbf{x} \leq b_i \right\} = \{ \mathbf{x} \in [0, 1]^n \mid A\mathbf{x} \leq \mathbf{b} \}.$$

Clearly, computing  $\text{Vol}(K_m)$  is  $\#P$ -hard. This section establishes the following theorem, extending the technique described in Sects. 2 and 3.

**Theorem 4.1** *For any  $\epsilon$  ( $0 < \epsilon \leq 1$ ), there exists an  $O((n^2\epsilon^{-1})^{m+1}nm \log m)$  time deterministic algorithm to approximate  $\text{Vol}(K_m)$  with approximation ratio  $1 + \epsilon$ .*

Notice that Theorem 4.1 claims  $O(n^{2m+3}/\epsilon^{m+1})$  time for any constant  $m$ .

As a preliminary step, we “normalize” the problem in a similar manner to Sect. 2. Let  $\tilde{A} = (\tilde{a}_{ij})$  be defined by

$$\tilde{a}_{ij} = \frac{a_{ij}}{b_i} M \quad (16)$$

for  $i = 1, \dots, m$  and  $j = 1, \dots, n$ , and also let  $\tilde{\mathbf{a}}_i^\top$  ( $i = 1, \dots, m$ ) and  $\tilde{A}_j$  ( $j = 1, 2, \dots, n$ ) respectively denote the row and column vectors of  $\tilde{A}$  (cf. (15)). Let

$$\tilde{K}_m = \left\{ \mathbf{x} \in [0, 1]^n \mid \tilde{A}\mathbf{x} \leq M\mathbf{1} \right\} = \left\{ \mathbf{x} \in [0, 1]^n \mid \bigwedge_{i=1, \dots, m} \tilde{\mathbf{a}}_i^\top \mathbf{x} \leq M \right\}$$

where  $\mathbf{1}$  is an  $m$ -dimensional vector defined by  $\mathbf{1} \stackrel{\text{def}}{=} (1, \dots, 1)^\top$ . It is not difficult to see that

$$\begin{aligned} K_m &= \left\{ \mathbf{x} \in [0, 1]^n \mid A\mathbf{x} \leq \mathbf{b} \right\} \\ &= \left\{ \mathbf{x} \in [0, 1]^n \mid \tilde{A}\mathbf{x} \leq M\mathbf{1} \right\} \\ &= \tilde{K}_m \end{aligned} \quad (17)$$

holds.

## 4.2 Convolution for $\text{Vol}(K_m)$

Now we explain a convolution for  $\text{Vol}(K_m)$ , in a similar manner to Sect. 2. Let  $\mathbf{X} = (X_1, \dots, X_n)$  be a uniform random variable over  $[0, 1]^n$ . Then, it is not difficult to see that  $[\tilde{A}\mathbf{X} \leq M\mathbf{1}]$  if and only if  $[\mathbf{X} \in \tilde{K}_m]$ , which implies that

$$\Pr[\tilde{A}\mathbf{X} \leq M\mathbf{1}] = \Pr[\mathbf{X} \in \tilde{K}_m] = \frac{\text{Vol}(\tilde{K}_m)}{\text{Vol}([0, 1]^n)} = \text{Vol}(\tilde{K}_m)$$

holds.

We define  $\Phi_0: \mathbb{R}^m \rightarrow \mathbb{R}$  by

$$\Phi_0(\mathbf{y}) = \begin{cases} 1 & \text{when } \mathbf{y} \geq \mathbf{0}, \\ 0 & \text{otherwise,} \end{cases}$$

for  $\mathbf{y} = (y_1, \dots, y_m)^\top$ . For  $j = 1, 2, \dots, n$ , we inductively define  $\Phi_j: \mathbb{R}^m \rightarrow \mathbb{R}$  by

$$\Phi_j(\mathbf{y}) = \int_{-\infty}^{\infty} \Phi_{j-1}(\mathbf{y} - s\tilde{A}_j) f(s) ds,$$

for  $\mathbf{y} = (y_1, \dots, y_m)^\top$ , where  $f$  is the uniform density function over  $[0, 1]$  (cf. (4)).

For convenience, let  $\mathbf{X}_{[j]} = (X_1, \dots, X_j)^\top$  for  $j \in \{1, \dots, n\}$ , as well as let  $\tilde{A}_{[j]}$  denote an  $m \times j$  submatrix of  $\tilde{A}$  defined by

$$\tilde{A}_{[j]} \stackrel{\text{def}}{=} (\tilde{A}_1 \quad \dots \quad \tilde{A}_j) = \begin{pmatrix} \tilde{a}_{1,1} & & \tilde{a}_{1,j} \\ \vdots & \dots & \vdots \\ \tilde{a}_{m,1} & & \tilde{a}_{m,j} \end{pmatrix}$$

for  $j \in \{1, \dots, n\}$ .

**Proposition 4.2** *For each  $j \in \{1, 2, \dots, n\}$ ,*

$$\Phi_j(\mathbf{y}) = \Pr \left[ \tilde{A}_{[j]} \mathbf{X}_{[j]} \leq \mathbf{y} \right]$$

*holds for any  $\mathbf{y} \in \mathbb{R}^m$ .*

*Proof* To begin with, consider the case of  $j = 1$ . Notice that  $\Phi_0$  is an indicator function in fact, then

$$\begin{aligned} \Phi_1(\mathbf{y}) &= \int_{-\infty}^{+\infty} \Phi_0(\mathbf{y} - s\tilde{A}_{[1]}) f(s) ds = \int_0^1 \Phi_0(\mathbf{y} - s\tilde{A}_{[1]}) ds \\ &= \int_0^1 \Phi_0 \left( \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} - \begin{pmatrix} s\tilde{a}_{1,1} \\ \vdots \\ s\tilde{a}_{m,1} \end{pmatrix} \right) ds = \Pr \left[ \tilde{A}_{[1]} \mathbf{X}_{[1]} \leq \mathbf{y} \right] \end{aligned}$$

where we remark that  $\mathbf{X}_{[1]} = (X_1)$ . We obtain the claim in the case.

Inductively assuming that  $\Phi_{j-1}(\mathbf{y}) = \Pr \left[ \tilde{A}_{[j-1]} \mathbf{X}_{[j-1]} \leq \mathbf{y} \right]$  holds, the case of  $j$  ( $j = 1, 2, \dots, n$ ) is given by

$$\begin{aligned} &\Pr \left[ \tilde{A}_{[j]} \mathbf{X}_{[j]} \leq \mathbf{y} \right] \\ &= \int_{-\infty}^{+\infty} \Pr \left[ \tilde{A}_{[j]} \mathbf{X}_{[j]} \leq \mathbf{y} \mid X_j = s \right] f(s) ds \\ &= \int_{-\infty}^{+\infty} \Pr \left[ \bigwedge_{i=1, \dots, m} [\tilde{a}_{i,1} X_1 + \dots + \tilde{a}_{i,j-1} X_{j-1} + \tilde{a}_{i,j} s \leq y_i] \right] f(s) ds \\ &= \int_{-\infty}^{+\infty} \Pr \left[ \bigwedge_{i=1, \dots, m} [\tilde{a}_{i,1} X_1 + \dots + \tilde{a}_{i,j-1} X_{j-1} \leq y_i - \tilde{a}_{i,j} s] \right] f(s) ds \end{aligned}$$

$$\begin{aligned}
 &= \int_{-\infty}^{+\infty} \Pr \left[ \tilde{A}_{\lfloor j-1 \rfloor} \begin{pmatrix} X_1 \\ \vdots \\ X_{j-1} \end{pmatrix} \leq \mathbf{y} - s \tilde{A}_j \right] f(s) ds \\
 &= \int_{-\infty}^{+\infty} \Phi_{j-1}(\mathbf{y} - s \tilde{A}_j) f(s) ds \\
 &= \Phi_j(\mathbf{y}).
 \end{aligned}$$

We obtain the claim.  $\square$

Let  $\Phi(M\mathbf{1})$  denote  $\Phi_n(M\mathbf{1})$ , for convenience. Then, Proposition 4.2, with (17) and (18), suggests the following.

**Corollary 4.3**  $\Phi(M\mathbf{1}) = \text{Vol}(K_m)$ .

### 4.3 Approximation of $\Phi$

Next, we explain an approximation of  $\Phi_j(\mathbf{y})$  by  $G_j(\mathbf{y})$ , in a similar manner to Sect. 3.1. We define  $G_0: \mathbb{R}^m \rightarrow \mathbb{R}_{\geq 0}$  by

$$G_0(\mathbf{y}) = \begin{cases} 1 & \text{when } \mathbf{y} > \mathbf{0}, \\ 0 & \text{otherwise.} \end{cases}$$

Inductively assuming  $G_{j-1}(\mathbf{y})$ , we define  $\overline{G}_j: \mathbb{R}^m \rightarrow \mathbb{R}_{\geq 0}$  for each  $j = 1, \dots, n$

$$\overline{G}_j(\mathbf{y}) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} G_{j-1}(\mathbf{y} - s \tilde{A}_j) f(s) ds = \int_0^1 G_{j-1}(\mathbf{y} - s \tilde{A}_j) ds \quad (18)$$

for  $\mathbf{y} = (y_1, \dots, y_m)^\top$ . Then, we define a staircase function  $G_j: \mathbb{R}^m \rightarrow \mathbb{R}_{\geq 0}$  by

$$G_j(\mathbf{y}) \stackrel{\text{def}}{=} \overline{G}_j(\lceil \mathbf{y} \rceil) \quad (19)$$

for  $\mathbf{y} \in \mathbb{R}^m$ , where  $\lceil \mathbf{y} \rceil = (\lceil y_1 \rceil, \dots, \lceil y_m \rceil)^\top$ . We remark the following fact.

**Proposition 4.4**

$$\overline{G}_j(\mathbf{y}) = \begin{cases} 0 & \mathbf{y} \notin \mathbb{R}_{>0}^m \text{ (i.e., if } \exists i' \text{ such that } y_{i'} \leq 0), \\ 1 & \mathbf{y} \geq \begin{pmatrix} \sum_{j'=1}^j \tilde{a}_{1,j'} \\ \vdots \\ \sum_{j'=1}^j \tilde{a}_{m,j'} \end{pmatrix}. \end{cases}$$

Here we explain the computation of  $G_j$ . Remark that it is sufficient to compute  $G_j(\mathbf{z})$  for  $\mathbf{z} \in \mathbb{Z}^m$ , because  $G_j(\mathbf{y}) = G_j(\lceil \mathbf{y} \rceil)$  holds for any  $\mathbf{y} \in \mathbb{R}^m$  by the definition (19) of  $G_j(\mathbf{y})$ . Let

$$T \stackrel{\text{def}}{=} \left\{ \mathbf{t} = \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix} \in \mathbb{R}_{\geq 0}^m \mid \begin{array}{l} \mathbf{t} = \mathbf{z} - s\tilde{A}_j \ (s \in [0, 1]), \text{ and} \\ \exists i \in \{1, \dots, m\}, t_i \in \mathbb{Z}_{>0} \end{array} \right\}.$$

Let  $\mathbf{t}^{(1)}, \mathbf{t}^{(2)}, \dots, \mathbf{t}^{(|T|)}$  denote the whole members in  $T$  such that  $\mathbf{t}^{(k)} < \mathbf{t}^{(k+1)}$  ( $k = 1, 2, \dots, |T| - 1$ ), i.e.,  $\mathbf{t}^{(k+1)} - \mathbf{t}^{(k)} = s\tilde{A}_j$  with an appropriate  $s > 0$  for any  $k = 1, 2, \dots, |T| - 1$ . For convenience, let  $\mathbf{t}^{(0)} = \mathbf{z} - \tilde{A}_j$  if  $\mathbf{z} - \tilde{A}_j \in \mathbb{R}_{>0}$ , otherwise  $\mathbf{t}^{(0)} = \mathbf{t}^{(1)}$ . We observe that

$$|T| \leq \|\mathbf{z} + \mathbf{1}\|_1 \quad (20)$$

holds, because each element  $t_i^{(k)}$  ( $i \in \{1, \dots, m\}$ ) of  $\mathbf{t}^{(k)}$  is monotone nondecreasing with respect to  $k$  ( $k = 1, 2, \dots, |T|$ ) for each  $i \in \{1, \dots, m\}$ , and  $\exists i \in \{1, \dots, m\}$ ,  $\lfloor t_i^{(k+1)} \rfloor \neq \lfloor t_i^{(k)} \rfloor$  holds for any  $k = 1, 2, \dots, |T| - 1$ . Remark that  $\mathbf{1}$  in (20) corresponds to members  $\mathbf{t} \in T$  satisfying that  $t_i = 0$  for some  $i \in \{1, \dots, m\}$ . Let

$$\begin{aligned} d_s(\mathbf{u}, \mathbf{v}) &\stackrel{\text{def}}{=} |\{s \in \mathbb{R} \mid \mathbf{u} \leq s\tilde{A}_j \leq \mathbf{v}\}| \\ &= \frac{v_1 - u_1}{\tilde{a}_{1,j}} \end{aligned} \quad (21)$$

for  $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$  such that  $\mathbf{v} = \mathbf{u} + s\tilde{A}_j$  for  $s \in [0, 1]$ . Then,

$$\begin{aligned} G_j(\mathbf{z}) &= \int_0^1 G_{j-1}(\mathbf{z} - s\tilde{A}_j) ds \\ &= \sum_{k=0}^{|T|} d_s(\mathbf{t}^{(k-1)}, \mathbf{t}^{(k)}) G_{j-1}(\mathbf{t}^{(k)}) \\ &= \sum_{k=0}^{|T|} d_s(\mathbf{t}^{(k-1)}, \mathbf{t}^{(k)}) G_{j-1}(\lceil \mathbf{t}^{(k)} \rceil) \end{aligned}$$

holds, where the last equality follows the definition (19) of  $G_{j-1}$ .

Thus, we see that the value of  $G_j(\mathbf{z})$  is computed from the values  $G_{j-1}(\mathbf{z}')$  ( $\mathbf{z}' \in \{1, 2, \dots, z\}^m$ ) and  $\tilde{a}_{1,j}$ , only. Observation (20) suggests that  $|T| = O(mM)$  when  $\mathbf{z} \in \{0, 1, 2, \dots, M\}^m$ . Thus, the sequence  $\mathbf{t}^{(1)}, \mathbf{t}^{(2)}, \dots, \mathbf{t}^{(|T|)}$  is found in  $O(mM \log m)$  time when  $\mathbf{z} \in \{0, 1, 2, \dots, M\}^m$ , e.g., using a merge sort in a wise way. Once we obtain the sequence,  $G_j(\mathbf{z})$  is computed in  $O(mM)$  time.

#### 4.4 Algorithm and Analysis

Based on the arguments in Sect. 4.3, our algorithm is described as follows.

**Algorithm 3** Input:  $\tilde{a}_1, \dots, \tilde{a}_m \in \mathbb{Q}_{\geq 0}^n$ .

1. Let  $G_0(\mathbf{y}) := 1$  for  $\mathbf{y} \geq 0$ , otherwise  $G_0(\mathbf{y}) := 0$ ;

2. For  $j = 1, \dots, n$
3. For each  $z \in \{0, 1, \dots, M\}^m$
4. Compute  $G_j(z)$ ;
5. Output  $G_n(M\mathbf{1})$ .

For convenience, let  $G(M\mathbf{1})$  denote  $G_n(M\mathbf{1})$ , in the following. Firstly, we discuss the time complexity of Algorithm 3.

**Lemma 4.5** *The running time of Algorithm 3 is  $O(nM^{m+1}m \log m)$ .*

*Proof* As stated in Sect. 4.3,  $G_j(z)$  is computed at line 4 in  $O(Mm \log m)$  time. Now, the claim is easy.  $\square$

The time complexity is possible to be improved in a similar manner to Lemma 3.3 (see also Sect. 3.4), but this paper omits the arguments.

Next, we establish the following approximation ratio of Algorithm 3.

**Lemma 4.6** *For any  $\epsilon$  ( $0 < \epsilon \leq 1$ ), set  $M \geq 2n^2\epsilon^{-1}$ , then we have*

$$\Phi(M\mathbf{1}) \leq G(M\mathbf{1}) \leq (1 + \epsilon)\Phi(M\mathbf{1}).$$

Theorem 4.1 is immediate from Lemmas 4.5 and 4.6. As a preliminary, we observe the following facts from the definitions of  $\bar{G}_j(\mathbf{y})$  and  $G_j(\mathbf{y})$ .

**Observation 4.7** *For any  $j \in \{1, 2, \dots, n\}$ ,  $\Phi_j(\mathbf{y})$ ,  $\bar{G}_j(\mathbf{y})$  and  $G_j(\mathbf{y})$  are monotone nondecreasing (with respect to  $\mathbf{y}$ ), respectively.*

**Observation 4.8** *For each  $j \in \{1, 2, \dots, n\}$ ,  $\bar{G}_j(\mathbf{y}) \leq G_j(\mathbf{y}) \leq \bar{G}_j(\mathbf{y} + \mathbf{1})$  holds for any  $\mathbf{y} \in \mathbb{R}^m$ .*

**Proposition 4.9** *For each  $j \in \{1, 2, \dots, n\}$ ,  $\Phi_j(\mathbf{y}) \leq \bar{G}_j(\mathbf{y})$  holds for any  $\mathbf{y} \in \mathbb{R}^m$ .*

*Proof* By the definition,  $\Phi_0(\mathbf{y}) = \bar{G}_0(\mathbf{y})$  for any  $\mathbf{y} \in \mathbb{R}^m$ . Inductively assuming that  $\Phi_{j-1}(\mathbf{y}) \leq \bar{G}_{j-1}(\mathbf{y})$  holds for any  $\mathbf{y} \in \mathbb{R}^m$ , we have that

$$\begin{aligned} \Phi_j(\mathbf{y}) &= \int_{-\infty}^{\infty} \Phi_{j-1}(\mathbf{y} - s\tilde{A}_j) f(s) ds \\ &\leq \int_{-\infty}^{\infty} \bar{G}_{j-1}(\mathbf{y} - s\tilde{A}_j) f(s) ds && \text{(Induction hypo.)} \\ &\leq \int_{-\infty}^{\infty} G_{j-1}(\mathbf{y} - s\tilde{A}_j) f(s) ds = \bar{G}_j(\mathbf{y}) && \text{(by Obs 4.8)} \end{aligned}$$

for any  $\mathbf{y} \in \mathbb{R}^m$ .  $\square$

Now, we establish the following Lemma 4.10, similar to Lemma 3.8.

**Lemma 4.10** *For each  $j \in \{1, \dots, n\}$ ,*

$$\Phi_j(\mathbf{y}) \leq G_j(\mathbf{y}) \leq \Phi_j(\mathbf{y} + j\mathbf{1})$$

*holds for any  $\mathbf{y} \in \mathbb{R}^m$ .*



*Proof* The former inequality is immediate from Proposition 4.9 and Observation 4.8. Now, we are concerned with the latter inequality. Observation 4.8 implies that

$$G_1(\mathbf{y}) \leq \overline{G}_1(\mathbf{y} + \mathbf{1}) = \Phi_1(\mathbf{y} + \mathbf{1}),$$

and obtain the claim for  $j = 1$ . Inductively assuming that  $G_{j-1}(\mathbf{y}) \leq \Phi_{j-1}(\mathbf{y} + (j-1)\mathbf{1})$  for any  $\mathbf{y} \in \mathbb{R}^m$ , we show that  $G_j(\mathbf{y}) \leq \Phi_j(\mathbf{y} + j\mathbf{1})$ . Notice that

$$\begin{aligned} \overline{G}_j(\mathbf{y}) &= \int_{-\infty}^{\infty} G_{j-1}(\mathbf{y} - s\tilde{A}_j) f(s) ds \\ &\leq \int_{-\infty}^{\infty} \Phi_{j-1}(\mathbf{y} - s\tilde{A}_j + (j-1)\mathbf{1}) f(s) ds \quad (\text{induction hypo.}) \\ &= \int_{-\infty}^{\infty} \Phi_{j-1}(\mathbf{y} + (j-1)\mathbf{1} - s\tilde{A}_j) f(s) ds \\ &= \Phi_j(\mathbf{y} + (j-1)\mathbf{1}) \end{aligned} \tag{22}$$

holds for any  $\mathbf{y} \in \mathbb{R}^m$ . Using Observation 4.8, (22) implies that

$$G_j(\mathbf{y}') \leq \overline{G}_j(\mathbf{y}' + \mathbf{1}) \leq \Phi_j((\mathbf{y}' + \mathbf{1}) + (j-1)\mathbf{1}) = \Phi_j(\mathbf{y} + j\mathbf{1})$$

and we obtain the claim.  $\square$

#### Corollary 4.11

$$\Phi(M\mathbf{1}) \leq G(M\mathbf{1}) \leq \Phi(M\mathbf{1} + n\mathbf{I}).$$

To transform the horizontal bound to a vertical one, we prepare the following.

#### Lemma 4.12

$$\frac{\Phi(M\mathbf{1})}{\Phi(M\mathbf{1} + n\mathbf{I})} \geq \left( \frac{M}{M+n} \right)^n.$$

*Proof* For convenience, let  $\tilde{K}(c) \stackrel{\text{def}}{=} \{\mathbf{x} \in [0, 1]^n \mid \tilde{A}\mathbf{x} \leq c\mathbf{1}\}$  for  $c \geq 0$ , i.e.,  $\Phi(M\mathbf{1}) = \text{Vol}(\tilde{K}(M))$  and  $\Phi(M\mathbf{1} + n\mathbf{I}) = \text{Vol}(\tilde{K}(M+n))$ . Let

$$D \stackrel{\text{def}}{=} \left\{ \lambda \mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} \in \tilde{K}, \lambda \in \mathbb{R} \text{ such that } 0 \leq \lambda \leq \frac{M+n}{M} \right\}.$$

Since  $\tilde{K}(M)$  is convex containing the origin  $\mathbf{0}$ , it is not difficult to see that  $D$  is similar to  $\tilde{K}(M)$  in the  $n$  dimensional space, which implies that

$$\frac{\tilde{K}(M)}{D} = \left( \frac{M}{M+n} \right)^n \tag{23}$$

holds. It may be helpful to consider a decomposition of  $\tilde{K}(M)$  into (truncated) cones for a confirmation of the above arguments.

Now, it is sufficient to prove that  $\tilde{K}(M+n) \subseteq D$ . Let  $\mathbf{u} \in \tilde{K}(M+n)$ , which implies that  $\tilde{A}\mathbf{u} \leq (M+n)\mathbf{1}$ . Let  $\mathbf{u}' = \frac{M}{M+n}\mathbf{u}$ , then

$$\tilde{A}\mathbf{u}' = \frac{M}{M+n}\tilde{A}\mathbf{u} \leq \frac{M}{M+n}(M+n)\mathbf{1} = M\mathbf{1}$$

holds. This implies that  $\mathbf{u} \in D$ .  $\square$

Lemma 4.6 is now easy from Corollary 4.11 and Lemma 4.12, in a similar way to the proof of Lemma 3.4.

## 5 Supplemental Proof

**Proposition 5.1** (*Proposition 3.1*)

$$\overline{G}_j(y) = \begin{cases} 0 & y \leq 0, \\ 1 & y \geq \sum_{j'=1}^j \tilde{a}_{j'}. \end{cases}$$

*Proof* The proof is an induction on  $j$ . Recall that  $G_0(y) = 0$  for  $y \leq 0$  and  $G_0(y) = 1$  for  $y > 1$ . Inductively assuming the claim holds for  $\overline{G}_{j-1}$ , we prove the claim for  $\overline{G}_j$ .

In case that  $y \leq 0$ , then  $y - \tilde{a}_j s \leq 0$  holds for  $s \geq 0$  since  $\tilde{a}_j \geq 0$ . Recall the definition (6), then  $G_{j-1}(y - \tilde{a}_j s) = \overline{G}_{j-1}(\lceil y - \tilde{a}_j s \rceil) = 0$  where the last equality follows the induction hypothesis. Thus,

$$\overline{G}_j(y) = \int_0^1 G_{j-1}(y - \tilde{a}_j s) \, ds = \int_0^1 0 \, ds = 0$$

holds. We obtain the claim in the case.

In case that  $y \geq \sum_{j'=1}^j \tilde{a}_{j'}$ , then  $y - \tilde{a}_j s \geq \sum_{j'=1}^{j-1} \tilde{a}_{j'}$  holds for  $s \leq 1$ . Recall the definition (6), then  $G_{j-1}(y - \tilde{a}_j s) = \overline{G}_{j-1}(\lceil y - \tilde{a}_j s \rceil) = 1$  where the last equality follows the induction hypothesis. Thus,

$$\overline{G}_j(y) = \int_0^1 G_{j-1}(y - \tilde{a}_j s) \, ds = \int_0^1 1 \, ds = 1$$

holds. We obtain the claim.  $\square$

**Acknowledgments** The authors would like to thank the anonymous reviewers for their valuable comments. This work is partly supported by Grant-in-Aid for Scientific Research on Innovative Areas MEXT Japan “Exploring the Limits of Computation (ELC)” (Nos. 24106008, 24106005).

## References

1. Ando, E., Kijima, S.: An FPTAS for the volume computation of 0-1 knapsack polytopes based on approximate convolution integral. *Lect. Notes Comput. Sci.* **8889**, 376–386 (2014)
2. Bandyopadhyay, A., Gamarnik, D.: Counting without sampling: asymptotics of the log-partition function for certain statistical physics models. *Random Struct. Algorithms* **33**, 452–479 (2008)
3. Bárány, I., Füredi, Z.: Computing the volume is difficult. *Discrete Comput. Geom.* **2**, 319–326 (1987)
4. Bayati, M., Gamarnik, D., Katz, D., Nair, C., Tetali, P.: Simple deterministic approximation algorithms for counting matchings. In: *Proceedings of STOC*, pp. 122–127 (2007)
5. Cousins, B., Vempala, S., Bypassing, K.L.S.: Gaussian Cooling and an  $O^*(n^3)$  Volume Algorithm. In: *Proceedings of STOC*, pp. 539–548 (2015)
6. Dyer, M.: Approximate counting by dynamic programming. In: *Proceedings of STOC*, pp. 693–699 (2003)
7. Dyer, M., Frieze, A.: On the complexity of computing the volume of a polyhedron. *SIAM J. Comput.* **17**(5), 967–974 (1988)
8. Dyer, M., Frieze, A., Kannan, R.: A random polynomial-time algorithm for approximating the volume of convex bodies. *J. Assoc. Comput. Mach.* **38**(1), 1–17 (1991)
9. Elekes, G.: A geometric inequality and the complexity of computing volume. *Discrete Comput. Geom.* **1**, 289–292 (1986)
10. Gamarnik, D., Katz, D.: Correlation decay and deterministic FPTAS for counting colorings of a graph. *J. Discrete Algorithms* **12**, 29–47 (2012)
11. Gopalan, P., Klivans, A., Meka, R.: Polynomial-time approximation schemes for knapsack and related counting problems using branching programs. [arXiv:1008.3187v1](https://arxiv.org/abs/1008.3187v1) (2010)
12. Gopalan, P., Klivans, A., Meka, R., Štefankovič, D., Vempala, S., Vigoda, E.: An FPTAS for #knapsack and related counting problems. In: *Proceedings of FOCS*, pp. 817–826 (2011)
13. Li, L., Lu, P., Yin, Y.: Approximate counting via correlation decay in spin systems. In: *Proceedings of SODA*, pp. 922–940 (2012)
14. Li, L., Lu, P., Yin, Y.: Correlation decay up to uniqueness in spin systems. In: *Proceedings of SODA*, pp. 67–84 (2013)
15. Li, J., Shi, T.: A fully polynomial-time approximation scheme for approximating a sum of random variables. *Oper. Res. Lett.* **42**, 197–202 (2014)
16. Lin, C., Liu, J., Lu, P.: A simple FPTAS for counting edge covers. In: *Proceedings of SODA*, pp. 341–348 (2014)
17. Lovász, L.: An algorithmic theory of numbers, graphs and convexity. *SIAM Society for Industrial and Applied Mathematics*, Philadelphia (1986)
18. Lovász, L., Vempala, S.: Simulated annealing in convex bodies and an  $O^*(n^4)$  volume algorithm. *J. Comput. Syst. Sci.* **72**, 392–417 (2006)
19. Štefankovič, D., Vempala, S., Vigoda, E.: A deterministic polynomial-time approximation scheme for counting knapsack solutions. *SIAM J. Comput.* **41**(2), 356–366 (2012)
20. Weitz, D.: Counting independent sets up to the tree threshold. In: *Proceedings of STOC*, pp. 140–149 (2006)