Annals of Operations Research (2019) 279:367–386
https://doi.org/10.1007/s10479-018-3111-9

ORIGINAL RESEARCH

# Approximation schemes for *r*-weighted Minimization Knapsack problems

**Khaled Elbassioni[1] · Areg Karapetyan[1] · Trung Thanh Nguyen[2]**

Published online: 6 December 2018


**Abstract**
Stimulated by salient applications arising from power systems, this paper studies a class of non-linear Knapsack problems with non-separable quadratic constrains, formulated in either binary or integer form. These problems resemble the duals of the corresponding variants of 2-weighted Knapsack problem (a.k.a., complex-demand Knapsack problem) which has been studied in the extant literature under the paradigm of smart grids. Nevertheless, the employed techniques resulting in a polynomial-time approximation scheme (PTAS) for the 2-weighted Knapsack problem are not amenable to its minimization version. We instead propose a greedy geometry-based approach that arrives at a quasi PTAS (QPTAS) for the minimization variant with boolean variables. As for the integer formulation, a linear programming-based method is developed that obtains a PTAS. In view of the curse of dimensionality, fast greedy heuristic algorithms are presented, additionally to QPTAS. Their performance is corroborated extensively by empirical simulations under diverse settings and scenarios.

**Keywords** Weighted Minimization Knapsack · Quasi polynomial-time approximation scheme · Polynomial-time approximation scheme · Power generation planning · Smart grid · Economic dispatch control

## 1 Introduction

This paper presents approximation schemes for a class of binary and integer non-linear Knapsack problems taking the forms

$$(r\text{-WBMKP}) \quad \min \quad \sum_{k=1}^{n} c_k x_k \tag{1}$$

✉ Trung Thanh Nguyen
ttnguyen.cs@gmail.com

Khaled Elbassioni
kelbassioni@masdar.ac.ae

Areg Karapetyan
akarapetyan@masdar.ac.ae

[1] Masdar Institute, Khalifa University of Science and Technology, Abu Dhabi, UAE

[2] Hai Phong University, Haiphong, Vietnam

$$\text{subject to} \quad \begin{array}{l} ||Ux||_2 \geq C, \\ x \in \{0, 1\}^n, \end{array} \tag{2}$$

and

$$(r\text{-WUMKP}) \quad \min \quad \sum_{k=1}^{n} c_k x_k \tag{3}$$

$$\text{subject to} \quad \begin{array}{l} ||Ux||_2 \geq C, \\ x \in \mathbb{Z}_+^n, \end{array} \tag{4}$$

where $r$ is a fixed constant, $n$ is the number of variables, $c^T = (c_1, \ldots, c_n) \in \mathbb{Q}_+^n$ is the cost vector, $U \in \mathbb{Q}_+^{r \times n}$, $C \in \mathbb{Q}_+$, and $||\cdot||_2$ denotes the Euclidean norm of vectors. Hereafter, these two problems are to be referred to as the $r$-weighted binary Minimization Knapsack problem ($r$-WBMKP) and the $r$-weighted unbounded Minimization Knapsack problem ($r$-WUMKP), respectively, in the order presented.

Evidently, both $r$-WBMKP and $r$-WUMKP are NP-hard as they accordingly generalize the Minimization Knapsack problem and its unbounded version (Kellerer et al. 2004). Yet, their ubiquitous applications underpin the routines critical for control and optimization of existing and emerging Alternating Current (AC) electric power networks (e.g. *Smart Grid*).[1] As such, 2-WBMKP (an instance of $r$-WBMKP where $r = 2$) arises as a subproblem in *Economic Dispatch* (ED) problem (a.k.a, Unit Commitment, Generation Dispatch Control) principal to power systems engineering (Wood and Wollenberg 2012). In ED for minimizing generation cost, there is typically a *load serving entity* (LSE) seeking to find optimal combination of power generations as to even demand and supply profiles without violating the operating constraints of a power system. Distinctively, $r$-WUMKP embodies the scenario of multiple generators of a type, where LSE is additionally required to specify the number of power sources to be dispatched for each type selected. Along these lines, the scalar $C$ in $r$-WBMKP and $r$-WUMKP represents the aggregate apparent power demand to be met, whereas the column vectors of matrix $U$ capture the active and reactive power outputs of contracted generation units. Accordingly, the vector $c$ encapsulates the costs of dispatching these units. Herein, the cost may quantify the perceived expenses to LSE accompanying utilization of a generation source, or alternatively, the amount being paid due to the $CO_2$ emissions.

*Related work* Basically, two major threads are evidenced in the line of research on non-linear Knapsack problems. The first group deals with the quadratic Knapsack problem which aims at maximizing a quadratic objective function subject to a linear constraint. A number of interesting approximation results have been proposed therein, by employing special structures of the graph underlying the objective function, such as *edge series-parallel graph* (Rader and Woeginger 2002), *bounded treewidth graphs* and *planar graphs* (Pferschy and Schauer 2013) or by considering *symmetric* quadratic objective of special forms (Kellerer and Strusevich 2010, 2012). The second direction is committed to problems where both objective functions and constraints are non-linear but *separable*, in a sense that the function can be presented as (or separated into) a sum of functions of single variables. These problems are also studied under a different name as *resource allocation problems*. A more comprehensive insight on

[1] In AC systems, the electric power is characterized by a complex number, where the real component is known as *active power* and the imaginary part reflects the *reactive power*. The combination of these two (i.e., magnitude of the complex-valued number) is called *apparent power*.

the results can be consulted in the book by Ibaraki and Katoh (1988) and in the survey by Bretthauer and Shetty (2002).

Notably, the first occurrence of a Knapsack problem with non-separable, non-linear constraints appeared in the work by Woeginger (2000), wherein the 2-weighted Knapsack problem was introduced. It was shown not to admit a fully polynomial-time approximation scheme (FPTAS), unless P = NP, via a reduction from the PARTITION problem (Garey and Johnson 1979). Very recently, Yu and Chau (2013) revisited the problem bridging its application to Smart Grid and provided a $\frac{1}{2}$-approximation algorithm. Later on, Karapetyan et al. (2018) obtained a greedy algorithm that attains a constant approximation factor only within $O(n \log n)$ time, $n$ standing for the input size, which was then extended in Khonji et al. (2016) for the scheduling variant of the problem. In Chau et al. (2014, 2016), a PTAS was given, consequently settling the complexity status of the problem. Elbassioni and Nguyen (2017) considered a general version of the 2-weighted Knapsack problem with linear and non-linear objective functions. As results, they proposed a PTAS for the linear objective function and the quadratic one of fixed nonnegative rank, and a $(\frac{1}{e} - \varepsilon)$-approximation algorithm for the submodular objective. The linear programming-based approach that attains the PTAS for the linear objective case in Elbassioni and Nguyen (2017) necessitates the polynomial solvability of the corresponding relaxation problem (where binary variables are relaxed to continuous variables between 0 and 1). In our case of $r$-WBMKP, however, a similar relaxation is unfortunately non-convex and it remains open whether it can be efficiently solved in polynomial time. On the other hand, it is interesting to either show the existence of a PTAS, or prove that $r$-WBMKP is APX-hard.

*Contributions* The key contribution of this study is centered on a theoretical proof of the existence of an approximation scheme for the $r$-weighted binary Minimization Knapsack problem ($r$-WBMKP), which has quasi-polynomial running time in the input size (see Sect. 2). Our theoretical result rules out the possibility that the problem is APX-hard, assuming NP $\not\subseteq$ DTIME($2^{polylog(n)}$). On the practical side, we propose two efficient heuristics guided by our theoretical results and their performance are evaluated through extensive simulations (see Sects. 4 and 5). Another contribution of this paper is to show a PTAS for the problem $r$-WUMKP with integer variables, by following a linear programming-based approach (see Sect. 3). Interestingly, although the corresponding relaxation of the problem (where nonnegative integer variables are replaced by nonnegative real ones) is still non-convex, yet a careful analysis can show that there exist optimal solutions which have only small number of non-zero components and thus can be efficiently computed by employing quantifier elimination algorithms (Basu 1999; Renegar 1992).

## Notation convention and preliminaries

We shall adhere to the following notation in the remainder of this paper. Let $[n]$ denote the set (or alternatively the range) $\{1, \ldots, n\}$. A vector $x \in \{0, 1\}^n$ is identified with a subset $S \subseteq [n]$, i.e, write $S = S(x) = \{j \in [n] \mid x_j = 1\}$. Set $\mathbf{0}$ and $\mathbf{1}$ to be the vectors of all zeros and ones, respectively. For a set $S \subseteq [n]$ and a set of vectors $\{q_1, \ldots, q_n\}$, write $q_S$ to denote the vectorial sum of all vectors $q_k, k \in S$. For any two vectors $x, y \in \mathbb{R}^n_+$, write $x \leq y$ if for every $j$th coordinates (or components) of $x$ and $y$, $j \in [n]$, $x_j \leq y_j$. We use $||x||_2$ to refer to the length (the Euclidean norm) of a vector $x \in \mathbb{R}^n$, i.e., $||x||_2 = \sqrt{x_1^2 + \cdots + x_n^2}$. Finally, define $|S|$ to be the cardinality of a set $S$.

For $\varepsilon \in (0, 1]$, a vector $x \in \{0, 1\}^n$ (resp., $x \in \mathbb{Z}_+^n$) is said to be *$\varepsilon$-optimal* for $r$-WBMKP (resp., $r$-WUMKP) if $x$ is a feasible solution satisfying $c^T x \leq (1 + \varepsilon) \cdot \textsc{Opt}$, where Opt is the value of an optimal solution. A PTAS is an algorithm that runs in time polynomial in the input size $n$, for every fixed $\varepsilon$, and outputs an $\varepsilon$-optimal solution. A QPTAS is similar to a PTAS but the running time is quasi polynomial (i.e., of the form $n^{\text{polylog}\, n}$), for every fixed $\varepsilon$.

## 2 A QPTAS for *r*-WBMKP

Let $\mathcal{I} = (c, U, C)$ be an instance of the $r$-WBMKP problem, and $\varepsilon > 0$ be a fixed constant. We will construct a quasi-polynomial-time algorithm which produces an $\varepsilon$-optimal solution to the instance $\mathcal{I}$. For simplicity, we write $c(S) := \sum_{k \in S} c_k$, for a set $S \subseteq [n]$. Recall that the linear programming-based method can not be applied here since the relaxed problem, which is obtained by considering $x_i \in [0, 1]$ for all $i \in [n]$, is *non-convex*, and thus we do not know if it can be solved efficiently (or at least can be approximated to within some constant factor). Instead, we will follow a geometric approach. We denote by Opt the value of an optimal solution to the instance $\mathcal{I}$. Our QPTAS is given as Algorithm 1. For simplicity, we assume that the algorithm has already a correct guess of the bound $B$ on the value of the optimal solution, that is, $B \leq \textsc{Opt} < (1 + \varepsilon)B$, where $B := (1 + \varepsilon)^i \min_k c_k$ for some $i \in \mathbb{Z}_+$. Note that the total number of possible guesses is $O(\log_{1+\varepsilon}(n \cdot \max_k c_k / \min_k c_k))$ and is thus polynomial in the size of the input. Given a correct guess of the bound $B$, we define

$$V := \{k \in [n] \mid c_k \geq (1 + \varepsilon)B\}, \quad \text{and} \quad T := \left\{k \in [n] \mid c_k < \frac{\varepsilon}{n} B\right\}. \tag{5}$$

We set $x_k = 1$ for all $k \in T$, and $x_k = 0$ for all $k \in V$, and assume therefore that we need to optimize over a set $N := [n] \backslash (T \cup V)$, for which $c_k \in [\frac{\varepsilon}{n} B, (1 + \varepsilon)B)$ for $k \in N$. Note that such restriction increases the cost of the solution obtained by at most $\varepsilon \cdot \textsc{Opt}$. To optimize over set $N$, the algorithm first decomposes the instance into classes according to utility and space (steps 1 and 3). The details of the decomposition will be described shortly. Next, the algorithm enumerates over all possible selections of a nonnegative integer $n_{s,l}$ associated to each region $s$ and a utility class $l$ (step 7); this number $n_{s,l}$ represents the largest length vectors that are taken in the potential solution from the set $N^s \cap N_l$. However, for technical reasons, the algorithm does this only for pairs $(s, l)$ for which the set $N^s \cap N_l$ contributes at least $\frac{1}{\varepsilon}$ in the optimal solution; the set of pairs that potentially do not satisfy this can be identified by enumeration (steps 5 and 6). To finish the description of the algorithm, it remains to formally describe the decomposition of $N$.

For the *utility partitioning*, we group the set of items in $N$ into $\ell := \lceil 1 + \log_{1+\varepsilon} \frac{n}{\varepsilon} \rceil$ (some possibly empty) *utility-classes* $N_1, \ldots, N_\ell$, where

$$N_l = \left\{k \in [n] \;\middle|\; c_k \in \left[\frac{\varepsilon}{n} \cdot B(1 + \varepsilon)^{l-1}, \frac{\varepsilon}{n} \cdot B(1 + \varepsilon)^l\right)\right\}, \tag{6}$$

for $l \in [\ell]$. Note that for all $k, k' \in N_l$, we have

$$c_k \leq c_{k'}(1 + \varepsilon). \tag{7}$$

The *space partitioning* is a bit more complicated and can be done as follows. For $k \in [n]$ define the vector $q_k \in \mathbb{Q}_+^r$ to be the $k$th column of $U$. Define the conic region $\mathcal{R}_T$ as follows.

$$\mathcal{R}_T \triangleq \left\{v \in \mathbb{R}_+^r : \; \|v\|_2 \leq C, \; v \geq q_T\right\}, \tag{8}$$

**Algorithm 1** QPTAS$(c, U, C)$

**Input:** A cost vector $c \in \mathbb{Q}_+^N$; accuracy parameter $\varepsilon$; matrix $U \in \mathbb{Q}_+^{r\times n}$; a scalar $C \in \mathbb{Q}_+$
**Output:** An $O(\sqrt{\varepsilon r})$-optimal solution $S$ to $r$-WBMKP
Obtain sets $T$ and $V$ as defined in (5) and set $N \leftarrow [n]\backslash(T \cup V)$
Let $\{q_k\}_{k\in N}$ be the vectors corresponding to the indices in $N$
Decompose the set $N$ into space-classes $N^1, \ldots, N^h$ and utility-classes $N_1, \ldots, N_\ell$
$S \leftarrow T \cup N$ ▷ Assume instance is feasible
**for** each subset of pairs $\mathcal{G} \subseteq [h] \times [\ell]$ **do**
  **for** each possible selection $(T_{s,l} \subseteq N^s \cap N_l : |T_{s,l}| \leq \frac{1}{\varepsilon}, (s,l) \in \mathcal{G})$ **do**
    **for** each possible selection $(n_{s,l} \in \{1, \ldots, |N^s \cap N_l|\} : (s,l) \in ([h] \times [\ell])\backslash\mathcal{G})$ **do**
      Let $S_{s,l}$ be the set of the $n_{s,l}$ vectors with largest length in $N^s \cap N_l$
      $S' \leftarrow T \cup \left(\bigcup_{(s,l)\in\mathcal{G}} T_{s,l}\right) \cup \left(\bigcup_{s,l} S_{s,l}\right)$
      **if** $\| \sum_{k\in S'} q_k \|_2 \geq C$ and $c(S') < c(S)$ **then**
        $S \leftarrow S'$
**return** $S$

where $q_T := \sum_{k\in T} q_k$. Then the problem now amounts to finding $S \subseteq N$ s.t. $q_S := \sum_{k\in S} q_k$ is not in the interior of $\mathcal{R}_T$. We now show how to partition the set of vectors (indices) in $N$ into $h = r(\frac{\sqrt{r}}{\varepsilon})^{r-1}$ *space-classes* $N^1, \ldots, N^h$, with the following property: for all $s \in [h]$, there exists $\xi(s) \in \mathbb{R}_+^r$ such that for all $k \in N^s$, the angle that any vector $q_k, k \in N^s$, makes with $\xi(s)$ is sufficiently small. In fact, it holds that

$$\frac{q_k^T \xi(s)}{\|q_k\|_2 \|\xi(s)\|_2} \geq 1 - \varepsilon. \tag{9}$$

We partition the region $\mathcal{R}_T$ into disjoint regions $\mathcal{R}_T(1), \ldots, \mathcal{R}_T(h)$, obtained as follows. For $j \in [r]$, we define $w_T^j$ as follows:

$$w_T^j := \sqrt{C^2 - \sum_{j'\neq j} \left(q_T^{j'}\right)^2} - q_T^j, \tag{10}$$

where $q_T^j$ denotes the $j$th coordinate of the vector $q_T$. Let $\mu = \bar{w}_T \cdot \mathbf{1}$, where $\bar{w}_T := \max_{j\in[r]} w_T^j$, and define the $r$-dimensional box

$$\mathcal{C}_T := \{\nu \in \mathbb{R}_+^r \mid q_T \leq \nu \leq q_T + \mu\}. \tag{11}$$

Note that $\mathcal{R}_T \subseteq \mathcal{C}_T$. We grid the $r$ facets of $\mathcal{C}_T$ that do not contain the point $q_T$ by interlacing equidistant $(r-2)$-dimensional parallel hyperplanes with inter-separation $\frac{\varepsilon \bar{w}_T}{\sqrt{r}}$, for each $j \in [r]$, and with the $j$th principal axis as their normal. Note that the grid corresponding to such a facet contains $(\frac{\sqrt{r}}{\varepsilon})^{r-1}$ cells. This makes the total number of grid cells is $h = r(\frac{\sqrt{r}}{\varepsilon})^{r-1}$; let us call them $\mathcal{C}_1, \ldots, \mathcal{C}_h$ (these are $(r-1)$-dimensional hypercubes). For each $s \in [h]$, we define the region $\mathcal{R}_T(s)$ as the $r$-dimensional simplex

$$\mathcal{R}_T(s) := \text{conv}(\{q_T\} \cup \mathcal{C}_s), \tag{12}$$

and denote by $\xi(s) = \kappa(s) - q_T$ the designated vector, where $\kappa(s)$ is the vertex center of the cell $\mathcal{C}_s$. Finally, we define

$$N^s := \{k \in [n] \mid q_T + q_k \in \mathcal{R}_T(s)\}, \tag{13}$$

for all $s \in [h]$. This gives the required space partitioning of the vectors. It remains to show that the angle condition (9) is satisfied. Indeed, consider any vector $q_k$ such that $q_T + q_k \in \mathcal{R}_T(s)$.

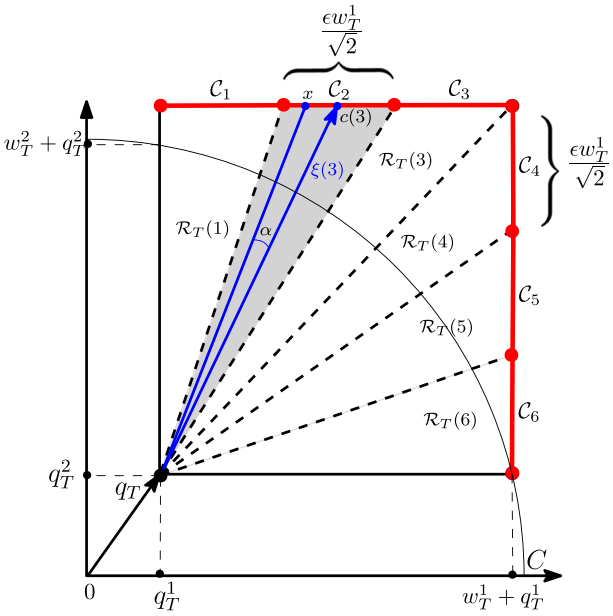


**Fig. 1** Partition procedure in the 2-dimensional space. The box $\mathcal{C}_T$ in this case is a square which has $q_T$ as one of its four vertices. The two red edges of the square, which do no contain the vertex $q_T$, are divided into equal line segments denoted by $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4, \mathcal{C}_5, \mathcal{C}_6$. Each line segment $\mathcal{C}_s$ together with the point $q_T$ form a triangle $\mathcal{R}_T(s)$

Let the ray $\{q_T + \lambda q_k : \lambda \geq 0\}$ hit the boundary cell $\mathcal{C}_s$ in the point $x$. Consider the triangle formed by the three points $q_T$, $\kappa(s)$ and $x$. Then, by construction, the following hold:

- the distances between $q_T$ and both $\kappa(s)$ and $x$ are at least $\bar{w}_T$ (the length of an edge of the $r$-dimensional box $\mathcal{C}_T$).
- the distance between $\kappa(s)$ and $x$ is at most $\frac{\varepsilon \bar{w}_T \sqrt{r-1}}{\sqrt{r}}$ (the length of a diagonal of the $(r-1)$-dimensional hypercube $\mathcal{C}_s$).

It follows that the angle $\theta$ between the two vectors $q_k$ and $\xi(s)$ is no more than

$$\sin^{-1}\left(\frac{\sqrt{r-1}}{\sqrt{r}}\varepsilon\right) \leq \sin^{-1}(\varepsilon),$$

implying $\cos(\theta) \geq \sqrt{1-\varepsilon} \geq 1-\varepsilon$ (9).

A more intuitive way to describe the space partitioning is to restrict our look at the 2-dimensional space (see Fig. 1). Given the quarter disk in the first quadrant of $\mathbb{R}^2$ with radius $C$ centered at the origin, and a nonnegative vector $q_T$ (which determines a point $q_T$ inside the quarter). We first define a square $\mathcal{C}_T$ of size $k$, where $k$ is the maximum of horizontal and vertical distances from $q_T$ to the boundary of the disk, and $\mathcal{C}_T$ has $q_T$ as one of its four vertices. The two red edges of the square, which do not contain the vertex $q_T$, are divided into a constant number of non-overlapping line segments of equal length, denoted by $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_h$, in such a way that the angle formed by the vertex $q_T$ and two rays connecting it to two endpoints of each line segment is small enough. Consequently, the square $\mathcal{C}_T$ is partitioned into $h$ convex regions (or more precisely, triangles) $\mathcal{R}_T(1), \mathcal{R}_T(2), \ldots, \mathcal{R}_T(h)$, which all share the common vertex $q_T$. Finally, the set $N$ is split into $h$ disjoint subsets

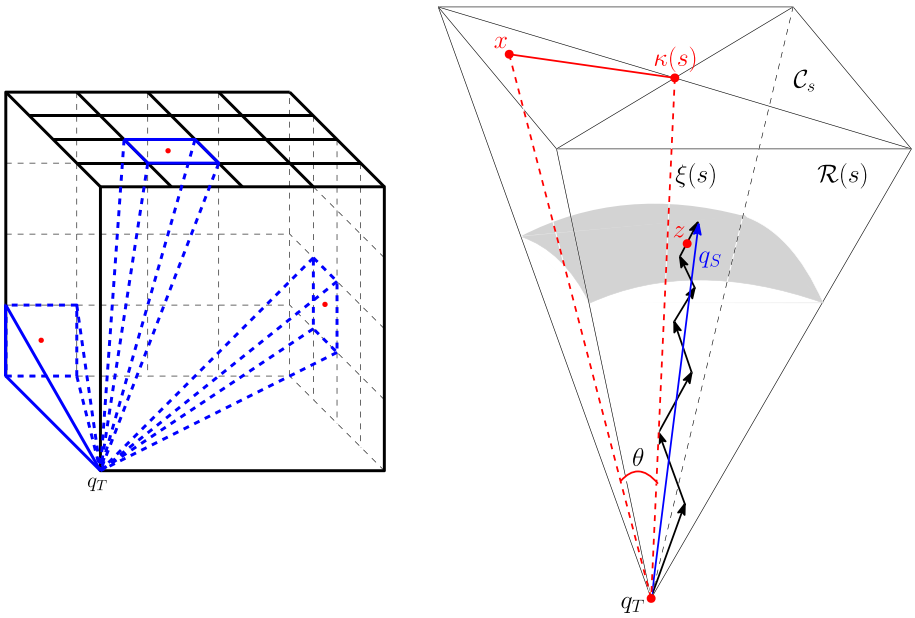


**Fig. 2** The figure on the left represents the partition of the box $\mathcal{C}_T$ into 3-dimensional simplexes in the three-dimension space. Three square facets of the box not containing the point $q_T$ are partitioned into small squares of the same sizes. The polyhedrons in blue color are examples of the convex hulls of $q_T$ and each of the small squares obtained by the partition. The figure on the right illustrates the polyhedron $\mathcal{R}(s)$, where $\kappa(s)$ is the center of the small square $\mathcal{C}(s)$. The shaded region represents the portion of the sphere of radius $C$ inside $\mathcal{R}(s)$. The vectors in black color illustrate the ones that are added into the greedy optimal solution $S$ in the Algorithm 2. The point $z$ is the intersection of the sphere and the last vector added into $S$

$N^1, N^2, \ldots, N^h$ according to this partition, namely, $N^s$ contains $k \in N$ for which the vector sum $q_T + q_k$ lies in the region $\mathcal{R}_T(s)$. An intuitive description of the space partitioning in the 3-dimensional space is given in Fig. 2.

We now state and prove the main result of this section.

**Theorem 1** *For any sufficiently small $\varepsilon > 0$, Algorithm 1 runs in time $n^{O((\frac{\sqrt{r}}{\varepsilon})^{r+1}\log n)}$ and outputs a solution $S$ satisfying $c(S) \le (1 + O(\sqrt{\varepsilon r}))\textsc{Opt}$ for the instance $\mathcal{I}$ of $r$-WBMKP.*

The following geometric Facts 1 and 2, and Lemma 1 are needed in the proof of Theorem 1. To be more focused, we only give here the proof of Lemma 1, and put the proofs of Facts 1 and 2 into "Appendix". Intuitively, Lemma 1 indicates that for a set of vectors $\{q_k : k \in N^s \cap N_l\}$ that lie in the same region and same utility-group, the *greedy* algorithm that processes the vectors in *non-increasing* order of length gives an $O(\varepsilon)$-optimal solution. In the proof of the lemma below, for simplicity we assume that $T = \varnothing$ and $q_T = \mathbf{0}$.

**Fact 1** *Let $a, b, \xi \in \mathbb{R}^r_+$ be such that $\frac{a^T\xi}{\|a\|_2\|\xi\|_2} \ge 1 - \varepsilon$ and $\frac{b^T\xi}{\|b\|_2\|\xi\|_2} \ge 1 - \varepsilon$. Then*

(i) $\dfrac{(a+b)^T\xi}{\|a+b\|_2\|\xi\|_2} \ge 1 - \varepsilon$, *and*

(ii) $\dfrac{a^Tb}{\|a\|_2\|b\|_2} \ge 1 - 5\varepsilon$.

Note that Fact 1(i) implies that, for any $S \subseteq N^s$, $q_S := \sum_{k\in S} q_k$ also satisfies the condition (9). Furthermore, for any two sets $S, S' \subseteq N^s$, we have

$$\frac{q_S^T q_{S'}}{\|q_S\|_2 \|q_{S'}\|_2} \geq 1 - 5\varepsilon.$$

**Fact 2** *Let $a, b \in \mathbb{R}^r_+$ be such that $\frac{a^T b}{\|a\|_2\|b\|_2} \geq 1 - 5\varepsilon$ and $\|\widehat{b}\|_2 = \lambda\|a\|_2$, where $\lambda \geq 1$ and $\widehat{b} := \mathtt{Pj}_a(b)$ is the projection of $b$ on $a$. Then for any vector $\eta \in \mathbb{R}^r_+$, it holds that*

$$\|\mathtt{Pj}_\eta(b)\|_2 \geq \lambda \left( \|\mathtt{Pj}_\eta(a)\|_2 - \frac{\sqrt{5\varepsilon(2-5\varepsilon)}}{1-5\varepsilon} \right) \|a\|_2.$$

**Algorithm 2** GREEDY- COVER$(c, \{q_k\}_{k\in N}, C)$

**Input:** A cost vector $c \in \mathbb{Q}^N_+$; accuracy parameter $\varepsilon$; vectors $q_k \in \mathbb{Q}^r_+$ satisfying (7) and (9); a scalar $C \in \mathbb{Q}_+$;
**Output:** $8\varepsilon$-optimal solution $S$ to $r$-WBMKP
1: $S' := \text{argmin}\{c(S) \mid S \subseteq N,\ |S| \leq \frac{1}{\varepsilon},\ q_S \text{ is feasible}\}$
2: Order the vectors $q_k, k \in N$, such that $\|q_1\|_2 \geq \|q_2\|_2 \geq \cdots$
3: $S \leftarrow \varnothing; k \leftarrow 0$
4: **while** $\|\sum_{k\in S} q_k\|_2 < C$ **do**
5: $\quad k \leftarrow k + 1$
6: $\quad S \leftarrow S \cup \{k\}$
7: **if** $c(S) \leq c(S')$ **then**
8: $\quad$ **return** $S$
9: **else**
10: $\quad$ **return** $S'$

**Lemma 1** *Consider instance of problem $r$-WBMKP described by a set of vectors $\{q_k\}_{k\in N}$ satisfying (7) and (9). Then, for any sufficiently small constant $\varepsilon > 0$, Algorithm 2 outputs a solution $S$ satisfying $c(S) \leq (1 + 8\varepsilon)$OPT.*

***Proof*** Let $S^*$ be an optimal solution. Since we consider every possible feasible solution of size at most $\frac{1}{\varepsilon}$ in step 1, we may assume w.l.o.g. that $|S^*| \geq \frac{1}{\varepsilon}$.

We claim that $|S| \leq \frac{|S^*|}{1-5\varepsilon} + 1$. To see this claim, let $q_{S^*} := \sum_{k\in S^*} q_k$, and for $k \in N$, denote by $\widehat{q}_k := \mathtt{Pj}_{q_{S^*}}(q_k)$ the projection of $q_k$ on $q_{S^*}$. Let $\nu \in S$ be the last vector added to $S$ in step 6 of the Algorithm 2. Then it is clear that

$$\sum_{k\in S\setminus\{\nu\}} \|\widehat{q}_k\|_2 < \|q_{S^*}\|_2, \tag{14}$$

since otherwise $S\setminus\{\nu\}$ would violate the condition in line 4 (this is because $\|\sum_{k\in S\setminus\{\nu\}} q_k\|_2 \geq \sum_{k\in S\setminus\{\nu\}} \|\widehat{q}_k\|_2 \geq \|q_{S^*}\|_2 \geq C$, by the feasibility of $S^*$). Define

$$X = (S\setminus\{\nu\}) \cap S^* \quad \text{and} \quad Y = (S\setminus\{\nu\})\setminus X.$$

Note that $X$ might be empty. We only consider here the case $X \neq \emptyset$ since the other case is similar. Inequality (14) implies that

$$\sum_{k\in S\setminus\{\nu\}} \|\widehat{q}_k\|_2 = \sum_{k\in X} \|\widehat{q}_k\|_2 + \sum_{k\in Y} \|\widehat{q}_k\|_2 < \|q_{S^*}\|_2 = \|\widehat{q}_{S^*\setminus X}\|_2 + \|\widehat{q}_X\|_2.$$

Since $\sum_{k\in X} \|\widehat{q}_k\|_2 = \|\widehat{q}_X\|_2$, we have

$$\sum_{k\in Y} \|\widehat{q}_k\|_2 < \|\widehat{q}_{S^*\setminus X}\|_2. \tag{15}$$

By the assumption that all the vectors $q_k, k \in N$ satisfy the angle condition (9), and by the Fact 1, it follows that $\|\widehat{q}_k\|_2 \geq (1-5\varepsilon)\|q_k\|_2$ for all $k \in N$. It follows by (15) that

$$\begin{aligned}(1-5\varepsilon)|Y| \min_{k\in Y} \|q_k\|_2 &\leq (1-5\varepsilon) \sum_{k\in Y} \|q_k\|_2 \leq \sum_{k\in Y} \|\widehat{q}_k\|_2 \\ &< \|\widehat{q}_{S^*\setminus X}\|_2 = \sum_{k\in S^*\setminus X} \|\widehat{q}_k\|_2 \leq \sum_{k\in S^*\setminus X} \|q_k\|_2 \\ &\leq |S^*\setminus X| \max_{k\in S^*\setminus X} \|q_k\|_2.\end{aligned}$$

Since $\max_{k\in S^*\setminus X} \|q_k\|_2 \leq \min_{k\in Y} \|q_k\|_2$ (by the greedy order in line 2 and the fact that $Y$ and $S^*\setminus X$ are disjoint sets), it follows that

$$|Y| \leq \frac{|S^*\setminus X|}{1-5\varepsilon} \tag{16}$$

By adding $|X|$ to both sides of the inequality above, we obtain

$$|X| + |Y| \leq |X| + \frac{|S^*\setminus X|}{1-5\varepsilon} = \frac{|S^*\setminus X| + |X| - 5\varepsilon|X|}{1-5\varepsilon} < \frac{|S^*|}{1-5\varepsilon} \tag{17}$$

Now the claim follows since $|S| = |S\setminus\{v\}| + 1 = |X| + |Y| + 1$.

By the utility condition (7),

$$\begin{aligned}c(S) \leq |S| \max_{k\in N} c_k \leq |S|(1+\varepsilon) \min_{k\in N} c_k &\leq \left(\frac{|S^*|}{1-5\varepsilon} + 1\right)(1+\varepsilon) \min_{k\in N} c_k \\ &= \left(\frac{1}{1-5\varepsilon} + \frac{1}{|S^*|}\right)(1+\varepsilon)|S^*| \min_{k\in N} c_k. \\ &\leq \left(\frac{1}{1-5\varepsilon} + \varepsilon\right)(1+\varepsilon)c(S^*).\end{aligned}$$

For sufficiently small $\varepsilon > 0$ we have $1/(1-5\varepsilon) < 1+6\varepsilon$, and thus

$$c(S) \leq (1+6\varepsilon)(1+\varepsilon)c(S^*) \leq (1+8\varepsilon)c(S^*).$$

The lemma follows. □

We are now ready to give a proof for Theorem 1.

***Proof of Theorem 1*** The running time is obvious since it is dominated by the number of selections in steps 5, 6 and 7, which is at most $\left(\frac{n}{h\ell}\right)^{O(\frac{h\ell}{\varepsilon})} = n^{O((\frac{\sqrt{r}}{\varepsilon})^{r+1}\log n)}$.

To see the approximation ratio, fix

$$\lambda := \left(1 - \frac{\sqrt{5\varepsilon(2-5\varepsilon)r}}{1-5\varepsilon}\right)^{-1}, \tag{18}$$

where we assume that $\varepsilon$ is chosen to be small enough to make sure that $\lambda$ is well defined (e.g., we can take $\varepsilon \in (0, 1/5)$).

Let $S^*$ be an optimal solution (within $N$), and for $s \in [h]$ and $l \in [\ell]$, let

$$S^*_{s,l} := S^* \cap N^s \cap N_l. \tag{19}$$

Define

$$\mathcal{G} := \left\{ (s,l) \in [h] \times [\ell] : \ |S^*_{s,l}| < \frac{1}{\varepsilon} \right\}, \tag{20}$$

and

$$T_{s,l} := S^* \cap N^s \cap N_l, \quad \text{for} \quad (s,l) \in \mathcal{G}, \tag{21}$$

and

$$\widehat{S} := S^* \setminus \left( \bigcup_{(s,l)\in\mathcal{G}} (N^s \cap N_l) \right). \tag{22}$$

For $k \in N^s \cap N_l$, let $\widehat{q}_k := \mathtt{Pj}_{q_{S^*_{s,l}}}(q_k)$ be the projection of $q_k$ on $q_{S^*_{s,l}} := \sum_{k\in S^*_{s,l}} q_k$. Define the set of pairs

$$\mathcal{H} := \left\{ (s,l) \in [h] \times [\ell] : \ \left\| \sum_{k\in N^s\cap N_l} \widehat{q}_k \right\|_2 \geq \lambda \|q_{\widehat{S}_{s,l}}\|_2 \text{ and } |S^*_{s,l}| \geq \frac{1}{\varepsilon} \right\}.$$

Then according to Lemma 1 (or more precisely, its proof), for every $(s,l) \in \mathcal{H}$ there is a choice $n_{s,l} \in \{1, \ldots, |N^s \cap N_l|\}$, such that if $S_{s,l}$ is the set of the $n_{s,l}$ vectors with largest length in $N^s \cap N_l$, then $\sum_{k\in S_{s,l}} \|\widehat{q}_k\|_2 \geq \lambda \|q_{\widehat{S}_{s,l}}\|_2$ and

$$c(S_{s,l}) \leq \left( \frac{\lambda}{1-5\varepsilon} + \varepsilon \right) (1+\varepsilon) c(\widehat{S}_{s,l}).$$

By this, we can define $p_k = \tau \cdot q_k$, for $\tau := \frac{\lambda \|q_{\widehat{S}_{s,l}}\|_2}{\sum_{k\in S_{s,l}} \|\widehat{q}_k\|_2} \leq 1$, such that $\sum_{k\in S_{s,l}} \|\widehat{p}_k\|_2 = \lambda \|q_{\widehat{S}_{s,l}}\|_2$, where $\widehat{p}_k := \mathtt{Pj}_{q_{\widehat{S}_{s,l}}}(p_k)$.

Let us now apply Fact 2 with $a = a_{s,l} := q_{\widehat{S}_{s,l}}$, $b = b_{s,l} := p_{S_{s,l}}$, and $\eta = q_{\widehat{S}}$ to get that

$$\|\mathtt{Pj}_\eta(b_{s,l})\|_2 \geq \lambda \left( \|\mathtt{Pj}_\eta(a_{s,l})\|_2 - \frac{\sqrt{5\varepsilon(2-5\varepsilon)}}{1-5\varepsilon} \right) \|a_{s,l}\|_2. \tag{23}$$

Summing the above inequalities for all $(s,l) \in \mathcal{H}$, we get

$$\sum_{(s,l)\in\mathcal{H}} \|\mathtt{Pj}_\eta(b_{s,l})\|_2 \geq \lambda \left( \sum_{(s,l)\in\mathcal{H}} \|\mathtt{Pj}_\eta(a_{s,l})\|_2 - \frac{\sqrt{5\varepsilon(2-5\varepsilon)}}{1-5\varepsilon} \sum_{(s,l)\in\mathcal{H}} \|a_{s,l}\|_2 \right) \tag{24}$$

Note that $\sum_{(s,l)\in\mathcal{H}} a_{s,l} = \eta$ by construction. By the Cauchy–Schwarz inequality and the nonnegativity of the vectors,

$$\begin{aligned} \left\| \sum_{(s,l)\in\mathcal{H}} \mathtt{Pj}_\eta(a_{s,l}) \right\|_2 = \|\eta\|_2 = \left\| \sum_{(s,l)\in\mathcal{H}} a_{s,l} \right\|_2 &\geq \frac{1}{\sqrt{r}} \left\| \sum_{(s,l)\in\mathcal{H}} a_{s,l} \right\|_1 \\ &= \frac{1}{\sqrt{r}} \sum_{(s,l)\in\mathcal{H}} \|a_{s,l}\|_1 \\ &\geq \frac{1}{\sqrt{r}} \sum_{(s,l)\in\mathcal{H}} \|a_{s,l}\|_2. \end{aligned}$$

Using this in (24), we obtain

$$\sum_{(s,l)\in\mathcal{H}} \|\mathtt{P}\mathtt{j}_\eta(b_{s,l})\|_2 \geq \lambda\left(1-\frac{\sqrt{5\varepsilon(2-5\varepsilon)r}}{1-5\varepsilon}\right)\sum_{(s,l)\in\mathcal{H}}\|\mathtt{P}\mathtt{j}_\eta(a_{s,l})\|_2 \tag{25}$$

$$\geq \sum_{(s,l)\in\mathcal{H}}\|\mathtt{P}\mathtt{j}_\eta(a_{s,l})\|_2, \tag{26}$$

by our choice of $\lambda$. From (24) and $q_{S_{s,l}} \geq b_{s,l}$, it follows by the feasibility of $S^*$ that the solution defined by $S = T \cup \left(\bigcup_{(s,l)\in\mathcal{G}} T_{s,l}\right) \cup \left(\bigcup_{s,l} S_{s,l}\right)$ is feasible.

Clearly, one of the choices in each of the enumeration steps 5, 6 and 7 will capture the above choices $\mathcal{G}$, $T_{s,l}$ for $(s,l) \in \mathcal{G}$, and $n_{s,l}$, for $(s,l) \in ([h]\times[\ell])\backslash\mathcal{G}$. It follows the procedure returns a solution $S$ with utility:

$$\begin{aligned}
c(S) &\leq c(T) + \sum_{(s,l)\in\mathcal{G}} c(T_{s,l}) + \sum_{(s,l)\notin\mathcal{G}} c(S_{s,l}) \\
&\leq c(T) + \sum_{(s,l)\in\mathcal{G}} c(T_{s,l}) + \left(\frac{\lambda}{1-5\varepsilon}+\varepsilon\right)(1+\varepsilon)\sum_{(s,l)\notin\mathcal{G}} c(\widehat{S}_{s,l}) \\
&\leq \left(\frac{1}{1-5\varepsilon-\sqrt{5\varepsilon(2-5\varepsilon)r}}+\varepsilon\right)(1+\varepsilon)\left(c(T) + \sum_{(s,l)\in\mathcal{G}} c(T_{s,l}) + \sum_{(s,l)\notin\mathcal{G}} c(\widehat{S}_{s,l})\right) \\
&\leq \left(\frac{1}{1-5\varepsilon-\sqrt{5\varepsilon(2-5\varepsilon)r}}+\varepsilon\right)(1+\varepsilon)c(S^*)
\end{aligned}$$

By an easy computation, we can prove that, for a sufficiently small constant $\varepsilon$, the constant factor in the inequality above can be bounded by $1+O(\sqrt{\varepsilon r})$. Thus, the claim in the theorem follows. □

## 3 A PTAS for *r*-WUMKP

This section presents a linear programming based approach that arrives at a PTAS for the $r$-weighted unbounded Minimization Knapsack problem ($r$-WUMKP). The approach is similar that of in Chandra et al. (1976) resulting in a PTAS for the integer multi-dimensional Knapsack problem. The basic idea is to find an optimal fractional solution to the relaxation of $r$-WUMKP, where integer variables are replaced by continuous ones, and then round up this solution to the nearest integer one. Note that the value of the fractional solution will be increased during the rounding procedure, and to keep this increase small a guess is required on the most expensive items in an optimal solution of $r$-WUMKP. Formally, the algorithm is described in Algorithm 3 followed by technical details on the proof of its approximation factor.

Let us denote by $p_i = (p_{ij})_{j=1}^n \in \mathbb{Q}_+^n, i \in [r]$, the row vectors of the matrix $U$, and rewrite the constraint (4) in the form

$$\sum_{i=1}^{r}(p_i^T x)^2 \geq C^2. \tag{27}$$

Also, let $\varepsilon > 0$ be a fixed constant, and define $\lambda = \lceil \frac{r}{\varepsilon} \rceil$. Without loss of generality, assume that $c_1 \geq c_2 \geq \cdots \geq c_n$, define $\mathcal{X}$ to be the set of vectors $s = (s_1, \ldots, s_n) \in \mathbb{Z}_+^n$ such that $\sum_{k=1}^n s_k \leq \lambda$, and let $X$ be the subset of vectors $s \in \mathcal{X}$ such that

**Algorithm 3** PTAS$(c, U, C)$

**Input:** A cost vector $c \in \mathbb{Q}_+^n$; the matrix $U \in \mathbb{Q}_+^{r\times n}$; a scalar $C$; and accuracy parameter $\varepsilon$
**Output:** A solution $v$ such that $c^T v \le (1+\varepsilon)$OPT
1: Order the $x_k$'s such that $c_1 \ge \cdots \ge c_n$
2: $\Omega \leftarrow \emptyset$
3: $\lambda \leftarrow \lceil \frac{r}{\varepsilon} \rceil$
4: $\mathcal{X} \leftarrow \{(s_1, \ldots, s_n) \in \mathbb{Z}_+^n | \sum_{k=1}^n s_k \le \lambda\}$
5: $X \leftarrow \{s \in \mathcal{X} | \sum_{i=1}^r (p_i^T s)^2 \le C^2\}$
6: $\Omega \leftarrow \Omega \cup (\mathcal{X} \backslash X)$
7: **for** each $s = (s_1, \ldots, s_n) \in X$ **do**
8: $\quad m \leftarrow \max\{k \in [n] | s_k \ne 0\}$
9: $\quad$ Compute an optimal solution $z$ to QP$(s)$, which has at most $r$ non-zero components
10: $\quad \overline{x} \leftarrow (s_1, \ldots, s_m + \lceil z_m \rceil, \lceil z_{m+1} \rceil, \ldots, \lceil z_n \rceil)$
11: $\quad \Omega \leftarrow \Omega \cup \{\overline{x}\}$
12: $v \leftarrow \text{argmin}_{x\in\Omega} \sum_{k=1}^n c_k x_k$
13: **return** $v$

$$\sum_{i=1}^{r} \left(p_i^T s\right)^2 \le C^2. \tag{28}$$

Observe that, by the definition of $X$, every vector $s \in \mathcal{X}\backslash X$ is a feasible solution to $r$-WUMKP. The size of $\mathcal{X}$ is bounded by $O(n^\lambda)$ and thus is polynomial in $n$ for every constant $\lambda$. For each $s \in X$, denote by $m$ the largest positive integer for which $s_m \ne 0$, and let

$$\Delta_i = p_i^T s, \quad \text{for } i \in [r].$$

If $s_k = 0$ for all $k = 1, \ldots, n$ then set $m = 0$. Consider the following quadratic programming problem.

$$\text{QP(S)} \quad \min \sum_{k=m}^{n} c_k x_k, \tag{29}$$

$$\text{subject to} \quad \sum_{i=1}^{r} \left(\Delta_i + \sum_{k=m}^{n} p_{ik} x_k\right)^2 \ge C^2, \tag{30}$$

$$x_k \in \mathbb{R}_+, \quad \text{for } k = m, \ldots, n. \tag{31}$$

Since QP$(s)$ is non-convex, the current techniques for efficiently solving convex programming cannot be applied. Fortunately, it is still plausible to show that one can find an optimal solution $z$ to this non-convex problem in polynomial time and, more importantly, $z$ has at most $r$ non-zero components (see Lemma 2 below). This solution $z$ is rounded up to its nearest integer, and then is combined with the vector $s$ to yield a feasible integer solution $\overline{x} = (s_1, \ldots, s_m + \lceil z_m \rceil, \lceil z_{m+1} \rceil, \ldots, \lceil z_n \rceil)$ to $r$-WUMKP.

Define $\Omega$ to be the set containing all such solutions $\overline{x}$ for all $s \in X$, and all the solutions $s \in \mathcal{X}\backslash X$. Then as a final solution we return the one among candidates in $\Omega$ with the smallest cost. The running time and correctness of the Algorithm 3 are given in the proof of Theorem 2 below.

**Lemma 2** *For every vector $s \in X$, one can find in polynomial time an optimal solution $z$ to* QP$(s)$ *such that $z$ has at most $r$ non-zero components.*

***Proof*** Fix $s \in X$ and let $z^*$ be an optimal solution to QP($s$). Define the following linear programming problem.

$$\overline{\text{QP}(s)} \quad \min \sum_{k=m}^{n} c_k x_k \tag{32}$$

$$\text{subject to} \quad \sum_{k=m}^{n} p_{ik} x_k \geq \sum_{k=m}^{n} p_{ik} z_k^*, \quad i \in [r] \tag{33}$$

$$x_k \in \mathbb{R}_+, \quad \text{for } k = m, \ldots, n. \tag{34}$$

It is obvious that $z^*$ is feasible to $\overline{\text{QP}(s)}$. Furthermore, every feasible solution to $\overline{\text{QP}(s)}$ is also feasible to QP($s$). Indeed, if $y$ is a feasible solution to $\overline{\text{QP}(s)}$, then

$$\sum_{i=1}^{r} \left( \Delta_i + \sum_{k=m}^{n} p_{ik} y_k \right)^2 \geq \sum_{i=1}^{r} \left( \Delta_i + \sum_{k=m}^{n} p_{ik} z_k \right)^2 \geq C^2.$$

The first inequality follows from the feasibility of $y$ for $\overline{\text{QP}(s)}$, while the second one follows from the feasibility of $z^*$ for QP($s$). Therefore, $z^*$ is also an optimal solution to $\overline{\text{QP}(s)}$. Moreover, any optimal solution to $\overline{\text{QP}(s)}$ is also an optimum to QP($s$) (but the reverse is not true). It is further noticed that since there are exactly $r$ non-trivial linear constraints in $\overline{\text{QP}(s)}$, there must exist an optimal solution $z$ which has at most $r$ non-zero components (see, e.g., Schrijver 1986; Nemhauser and Wolsey 1999). To find such an optimal solution, one needs to consider $O(n^r)$ possibilities of choosing $r$ non-zero variables among $n - m + 1$ variables $x_m, x_{m+1}, \ldots, x_n$. For each choice of variables the resulting problem can be solved in time polynomial in $r$ by using quantifier elimination algorithms (see Basu 1999; Renegar 1992). □

**Theorem 2** *For any fixed $\varepsilon > 0$, Algorithm 3 runs in polynomial time and produces an $\varepsilon$-optimal solution to the input instance.*

***Proof*** The running time of Algorithm 3 is dominated by the for-loop which has $|X| = O(n^\lambda) = O(n^{\lceil \frac{r}{\varepsilon} \rceil})$ iterations. Each of these iterations requires computing an optimal solution to a non-convex quadratic programming in fixed dimension $r$, which is known to be done in polynomial time. Therefore, the overall running time of Algorithm 3 is polynomial in the size of the input, for any fixed constant $\varepsilon$.

We now argue that the solution $v$ returned by the algorithm is $\varepsilon$-optimal. Indeed, let $x^*$ be an optimal solution to $r$-WUMKP of cost $c^T x^* = \text{OPT}$. If $\lambda \geq \sum_{k=1}^{n} x_k^*$, then $v$ is exactly an optimum to the $r$-WUMKP. Suppose that $\lambda < \sum_{k=1}^{n} x_k^*$. Let $s = (s_1, \ldots, s_m, \ldots, s_n)$ be the vector such that

- $\sum_{k=1}^{n} s_k = \lambda$,
- $s_k = x_k^*$ for all $k = 1, \ldots, m-1$,
- $s_m \neq 0$ and $s_k = 0$ for all $k = m+1, \ldots, n$.

Apparently, $s \in X$ and Eq. (28) holds, and hence an optimal solution $z$ for QP($s$) is found in step 9 of the algorithm.

Let $\overline{x} \in \mathbb{Z}_+^n$ be the rounded solution obtained from $z$ and $s$ in Step 10. Then

$$\sum_{k=1}^{n} c_k \overline{x}_k = \sum_{k=1}^{m-1} c_k \overline{x}_k + \sum_{k=m}^{n} c_k \overline{x}_k. \tag{35}$$

Furthermore, since $z$ has at most $r$ non-zero components, it follows that

$$\sum_{k=m}^{n} c_k \overline{x}_k \leq c_m s_m + \sum_{k=m}^{n} c_k z_k + r \cdot \max_{k=m}^{n} c_k.$$

Note that $c_1 \geq \cdots \geq c_n$, and thus

$$r \cdot \max_{k=m}^{n} c_k = r c_m = r c_m \cdot \frac{1}{\lambda} \sum_{k=1}^{m} s_k \leq \varepsilon \sum_{k=1}^{m-1} c_k \overline{x}_k + \varepsilon c_m s_m,$$

since

$$\lambda = \sum_{k=1}^{n} s_k = \sum_{k=1}^{m} s_k.$$

Finally, Eq. (35) implies that

$$\begin{aligned}
\sum_{k=1}^{n} c_k \overline{x}_k &\leq \sum_{k=1}^{m-1} c_k \overline{x}_k + c_m s_m + \varepsilon \sum_{k=1}^{m-1} c_k \overline{x}_k + \varepsilon c_m s_m + \sum_{k=m}^{n} c_k z_k \\
&\leq (1+\varepsilon) \left( \sum_{k=1}^{m-1} c_k \overline{x}_k + c_m s_m + \sum_{k=m}^{n} c_k z_k \right) \\
&\leq (1+\varepsilon) \left( \sum_{k=1}^{m-1} c_k \overline{x}_k + \sum_{k=m}^{n} c_k x_k^* \right) \\
&= (1+\varepsilon) \textsc{Opt}.
\end{aligned}$$

By the definition of $v$ and the fact that $\overline{x} \in \Omega$, we must have that $c^T v \leq c^T \overline{x} \leq (1+\varepsilon)\textsc{Opt}$. This completes the proof. □

## 4 Greedy heuristic algorithms for *r*-WBMKP

Despite its theoretical appeal, the proposed QPTAS imposes profound computational burdens, thereby impairing its practical credibility. In fact, invoking the QPTAS even on small instances of $r$-WBMKP problem might spell intractability. Thus, for the sake of practicality, this section presents two computationally efficient greedy heuristics, Greedy Relative Cost (GRC) and Greedy Geometric Search (GGS), given in Algorithms 4 and 5, respectively. The subsequent section carries out comprehensive numerical experiments to draw further insights on, and therein validate, their performance.

As such, GRC is a direct descendant of a greedy 2-approximation algorithm, introduced in Csirik et al. (1991), for the conventional 0–1 Minimization Knapsack problem. Once executed, it starts by sorting the items $\{q_k\}_{k\in[n]}$ according to non-decreasing order of their cost-to-magnitude ratio (a.k.a., relative cost). Then, while scanning through the items sequentially in that order, the algorithm examines a set of candidate solutions one selection at a time whenever feasible, in a sense detailed in Algorithm 4. Upon termination, GRC outputs the one amongst candidate solutions with the minimum cost.

Explained in Algorithm 5, GGS is a computationally savvy spin-off from the QPTAS that sacrifices its optimality in favor of tractability as a ramification of inhibiting certain,

**Algorithm 4** Greedy Relative Cost$(c, \{q_k\}_{k\in[n]}, C)$

**Input:** A cost vector $c \in \mathbb{Q}_+^n$; vectors $q_k \in \mathbb{Q}_+^r$; a scalar $C \in \mathbb{Q}_+$

Sort items in $[n]$ in non-decreasing order of their relative costs such that if $j \preceq j'$, then $\frac{c_j}{\|q_j\|_2} \leq \frac{c_{j'}}{\|q_{j'}\|_2}$, for $\forall\, j, j' \in [n]$ and denote this order by $E$
$S \leftarrow [n]$; $S' \leftarrow \emptyset$
**for** $k \in E$ (in the sorted order) **do**
  **if** $\|\sum_{j\in S'} q_j + q_k\|_2 < C$ **then**
    $S' \leftarrow S' \cup \{k\}$
  **else**
    **if** $c(S' \cup \{k\}) < c(S)$ **then**
      $S \leftarrow S' \cup \{k\}$
**return** $S$

**Algorithm 5** Greedy Geometric Search$(c, \{q_k\}_{k\in[n]}, h, C)$

**Input:** A cost vector $c \in \mathbb{Q}_+^n$; vectors $q_k \in \mathbb{Q}_+^r$; an integer $h \in \mathbb{Z}_+$; a scalar $C \in \mathbb{Q}_+$
Partition vectors in $\{q_k\}_{k\in[n]}$ into $h$ disjoint space-classes $N_1, \ldots, N_h$ as explained in Sect. 2
Sort items $\{q_k\}_{k\in N_i}$ in each class $N_i$, $i \in [h]$ in non-increasing order of their magnitudes
$S \leftarrow [n]$
**for** $s \in [h]$ **do**
  **for** each possible selection $\left(T \subseteq \bigcup_{i\in[h]} N_i : i \neq s\right)$ **do**
    $S' \leftarrow \{k\}_{k\in N_s} \cup \{k\}_{k\in T}$
    **if** $\|\sum_{k\in S'} q_k\|_2 \geq C$ and $c(S') < c(S)$ **then**
      $S \leftarrow S'$
**return** $S$

otherwise important, operations from the inherited kernel. Concretely, the partitioning subroutine of QPTAS is reduced in dimensionality to space classes only, with their number being determined by an input parameter $h$ of GGS. Following steps analogous to those in QPTAS, after sorting the items in each of the $h$ classes by their magnitudes in a descending order, the algorithm enumerates over all possible selections of $n_i$ vectors with the largest magnitudes in each class $i \in [h]$. Lastly, it delivers as a solution by far the most parsimonious feasible selection identified during the heuristic search. Nevertheless, the notably alleviated computational overhead comes at an inevitable cost of diminished solution quality leaving GGS devoid of any non-trivial approximation guarantees.

## 5 Performance evaluation

This section appraises performance of the presented greedy algorithms GRC and GGS through extensive empirical analysis on a simulated LSE system under diverse settings and scenarios. The objective values of the two algorithms for $r$-WBMKP problem are benchmarked with that of optimal solution computed numerically by APOPT (Hedengren 2014) optimizer. It is noteworthy that most of the available solvers, including Gurobi and CPLEX failed to tackle $r$-WBMKP problem properly in a reasonable time frame, even for small-scale input instances. As for the APOPT optimizer, the observed superiority in performance over the alternatives was conditional on the input dimensionality of $r$-WBMKP being merely bounded. This iterates the necessity for greater commitment towards devising efficient approximations for the problems under study, thereby embracing the effort put forth in the present work.

The following subsections detail the simulation setup and lay out the results of case studies performed.

### 5.1 Simulation setup and settings

The simulated model envisions an LSE serving a number of consumers each having certain apparent power demand requirements. In aggregate, the demand to be satisfied on the load side (i.e., $C$) is held constant at a level of 1MVA. It is assumed that the generation on LSE encompasses a hybrid mixture of conventional and renewable energy supplies alongside with power storage devices that total in number from 200 to 700. Each of these sources is assigned a specific power output level (i.e., $q_k$ which includes both active and reactive power generation amounts) and a generation cost (i.e., $u_k$) determined according to a probability preference model. To account for purely active/reactive power feed-ins and ensure properly functioning supply side on LSE, the power output vector of a committed generation unit is restricted to lie in the non-negative orthant of the plane (i.e., $q_k \geq \mathbf{0}$ for $\forall\, k \in [n]$).

A number of case studies are performed to evaluate the featured greedy algorithms considering various scenarios pertaining to power supplies' output profile and its correlation to generation costs. In particular, the following are settings for the case studies under study.

(i) *Generation cost function*:
(a) *Quadratic cost* (***Q***): The generation cost of a supply $k \in [n]$ is a quadratic function of its power output magnitude in the form of $c_k(\|q_k\|_2) = a \cdot \|q_k\|_2^2 + b \cdot \|q_k\|_2 + c$, where $a > 0, b, c \geq 0$ are preset constants.
(b) *Random setting* (***R***): The cost of a source is independent of its power generation magnitude and takes values drawn at random.
(c) *Uniform cost* (***U***): All the generation units, regardless of their power output potential, induce the same cost.

(ii) *Generation profiles*:
(a) *Small-scale power supplies* (***S***): The system is fed solely by sources having small power outputs ranging from 3KVA to 15KVA.
(b) *Mixed generation* (***M***): The generation set comprises a combination of both small and large scale power resources. The latter though constitute no more than 20% of the set chosen at random, capacitate sizable generation volumes ranging from 300KVA up to 1MVA .

In the sequel, a case study will be typified by the corresponding acronyms listed above. For example, the case study named QS stands for the one with small-scale power supplies and quadratic generation costs.

### 5.2 Results and discussion

Using the simulation model established, the candidate algorithms GRC and GGS are compared here in terms of quality of solution and running time for $r$-WBMKP problem. The optimal solutions computed by APOPT optimizer serve as a base case for the comparison. The results are summarised in Figs. 3 and 4, which depict outcomes of the two algorithms for various case studies, each being analyzed considering changes in the generation set. Specifically, GRC and GGS are applied 20 times for each of the $m$ number of generation sources, $m$ varying between 200 to 700 in steps of 50, in every case study. Moreover, each of these 20 iterations yields random perturbations in power output magnitudes and costs of

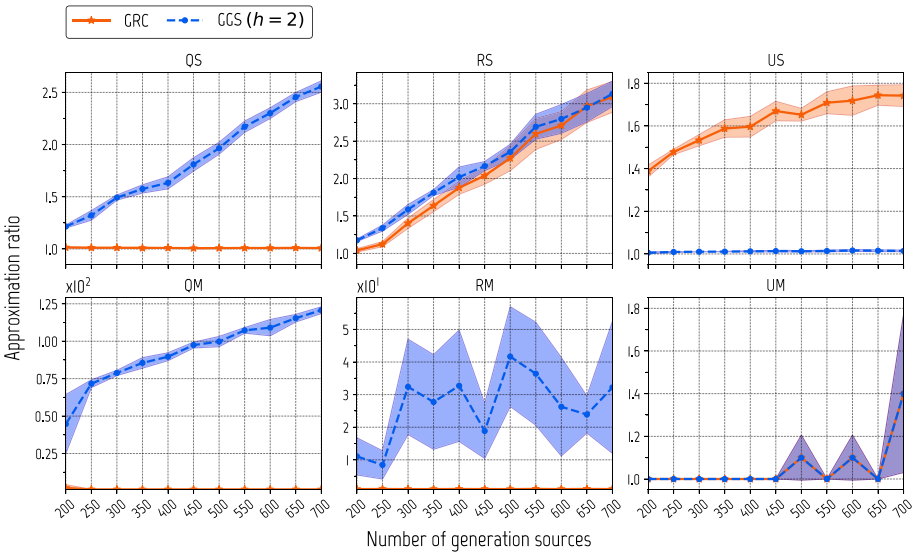


**Fig. 3** The average approximation ratios of GRC and GGS for $r$-WBMKP problem against the number of generation sources at 95% confidence interval

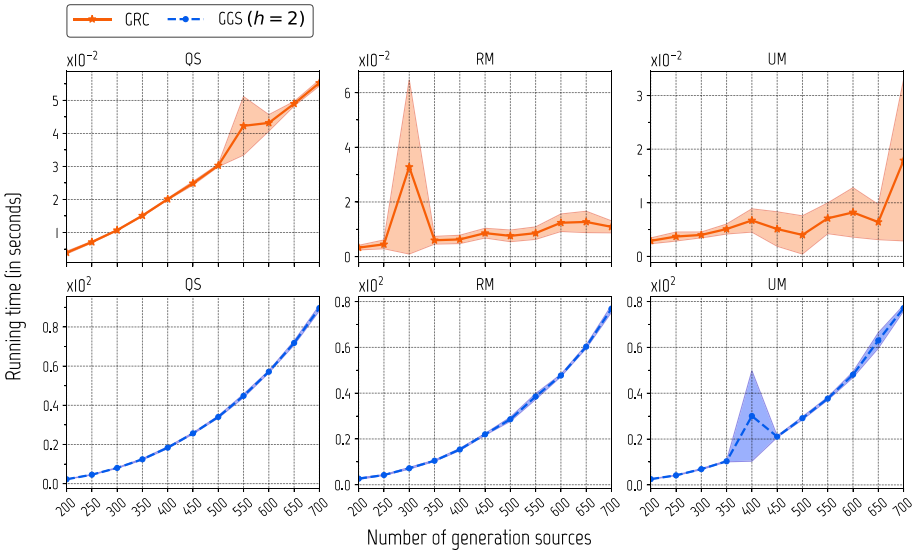


**Fig. 4** The average running time of GRC and GGS for $r$-WBMKP problem against the number of generation sources at 95% confidence interval

generation units. For clarity of presentation, the results pertinent to only one choice of the input parameter $h$ of GGS are reported.

When it comes to running time, GRC is uniquely preferable to GGS as suggested by Fig. 4 and the fact that it entails $O(n \log n)$ time to terminate, whereas GGS executes in

$O(n^h + n \log n)$ time.[2] This is not the case, however, from the perspective of their solutions' optimality extent. Indeed, the observed performance of the two greedy algorithms is rather complementary than concurrent, in a sense that either of them might fail to efficiently minimize the generation cost, but not both. In case studies with quadratic and random costs, which appear in Fig. 3, GGS drifts far away from the vicinity of optimal solutions approaching nearly 120-approximation in some cases, as opposed to GRC having 3.5-approximation in the worst-case. On the other hand, GGS proves superior to the latter for case studies with uniform costs, meanwhile manifesting near-optimal performance. These results are attributed primarily to the design of GGS, which heuristically selects the largest power sources first, essentially abstracting away their generation costs.

Importantly, the conducted numerical analysis alludes to the benefits expected to accrue from the confluence of the two greedy strategies. This could be achieved by replacing the sorting criteria in GGS with that of employed in GRC (i.e., by the non-decreasing order of relative costs). Premising that the combined variant inherits the best performance of those two algorithms, it could pave the foundations for deriving constant-factor approximations for $r$-WBMKP problem.

## 6 Conclusion

Motivated by applications in power systems, we have considered in this paper the two variants of non-linear Knapsack problems, $r$-WBMKP and $r$-WUMKP, one with binary variables and another with integer variables. As the results, we have obtained a PTAS for the second problem, whilst for the first one, which seems to be harder due to the non-convexity of the corresponding relaxation problem, only a QPTAS have been presented. On the theoretical side, our QPTAS is an important result toward the development of a PTAS for the first problem. On the practical side, since the worst case running time of the QPTAS is $n^{O(\log n)}$ (for constant $r$ and $\varepsilon$), this makes the algorithm impractical when the number of variables is large. Therefore, two heuristic algorithms with faster running times, GLC and GGS, have been provided. Furthermore, both of them have been verified to be efficient in practice through various experiments. For future work, it would be interesting to develop practical algorithms with provable guarantees on the distance of the returned solution to the optimal one.

**Acknowledgements** We would like to thank the Editor and anonymous reviewers for their careful reading of our manuscript and their helpful comments that improved the presentation of the paper.

## Appendix

***Proof of Fact 1*** (i) Indeed,

$$\begin{aligned}\frac{(a+b)^T\xi}{\|a+b\|_2\|\xi\|_2} &= \frac{\|a\|_2}{\|a+b\|_2}\cdot\frac{a^T\xi}{\|a\|_2\|\xi\|_2} + \frac{\|b\|_2}{\|a+b\|_2}\cdot\frac{b^T\xi}{\|b\|_2\|\xi(s)\|_2}\\ &\geq \frac{\|a\|_2+\|b\|_2}{\|a+b\|_2}\cdot(1-\varepsilon)\\ &\geq 1-\varepsilon,\end{aligned}$$

by the triangular inequality.

[2] Note that some of the case studies are omitted from Fig. 4 due to their close resemblance to the plotted ones.

(ii) Let $\bar{a} := \frac{a}{\|a\|_2}$, $\bar{b} := \frac{b}{\|b\|_2}$, and $\bar{\xi} := \frac{\xi}{\|\xi\|_2}$. If $a, b, \xi$ all lie in the same two-dimensional subspace, then the claim follows since the angle between $a$ and $b$ is no more than the sum of the angles between $a$ and $\xi$, and $b$ and $\xi$, which is at most is $\cos^{-1}((1-\varepsilon)^2 - \varepsilon(2-\varepsilon)) \leq \cos^{-1}(1-4\varepsilon)$. Otherwise, let $\bar{b} = \widehat{b} + \tilde{b}$ be the orthogonal decomposition of $\bar{b}$ with respect to the 2-dimensional subspace formed by the two vectors $\bar{a}$ and $\bar{\xi}$, where $\widehat{b}$ is the projection of $\bar{b}$ into this space, and $\tilde{b}$ is the orthogonal component. Then

$$\frac{\widehat{b}^T \bar{\xi}}{\|\widehat{b}\|_2} = \frac{\bar{b}^T \bar{\xi}}{\|\widehat{b}\|_2} \geq \frac{1-\varepsilon}{\|\widehat{b}\|_2} \geq 1-\varepsilon,$$

(since $\|\widehat{b}\|_2 \leq \|\bar{b}\|_2 = 1$), which also implies that $\|\widehat{b}\|_2 \geq 1-\varepsilon$. Since $a, \widehat{b}$, and $\xi$ lie in the same subspace, it follows by the above argument that $\frac{\widehat{b}^T \bar{a}}{\|\widehat{b}\|_2} \geq (1-4\varepsilon)$, and hence,

$$\bar{b}^T \bar{a} = \widehat{b}^T \bar{a} \geq (1-4\varepsilon)\|\widehat{b}\|_2 \geq (1-4\varepsilon)(1-\varepsilon),$$

implying the claim. □

***Proof of Fact 2*** Since the statement is invariant under rotation, we may assume, w.l.o.g., that $\eta = \mathbf{1}_j$, the $j$th-dimensional unit vector in $\mathbb{R}^r$. Write $b := \widehat{b} + \tilde{b}$, where $\tilde{b}$ is the vector orthogonal to $a$ in the subspace spanned by $a$ and $b$. Then $\|\mathtt{Pj}_\eta(b)\|_2 = b^j$ is the $j$th component of $b$, and $\|\mathtt{Pj}_\eta(\widehat{b})\|_2 = \widehat{b}^j$. Since

$$\|\widehat{b}\|_2 \geq \frac{a^T \widehat{b}}{\|a\|_2} = \frac{a^T b}{\|a\|_2} \geq (1-5\varepsilon)\|b\|_2,$$

it follows that

$$\|\tilde{b}\|_2 = \sqrt{\|b\|_2^2 - \|\widehat{b}\|_2^2} \leq \sqrt{5\varepsilon(2-5\varepsilon)}\|b\|_2 \leq \frac{\sqrt{5\varepsilon(2-5\varepsilon)}}{1-5\varepsilon}\|\widehat{b}\|_2 = \frac{\lambda\sqrt{5\varepsilon(2-5\varepsilon)}}{1-5\varepsilon}\|a\|_2.$$

Since $|b^j - \widehat{b}^j| = |\tilde{b}^j| \leq \|\tilde{b}\|_2$, and $\widehat{b} = \lambda\|a\|_2 \cdot a$, the claim follows. □

## References


Basu, S. (1999). New results on quantifier elimination over real closed fields and applications to constraint databases. *Journal of the ACM*, *46*(4), 537–555.

Bretthauer, K. M., & Shetty, B. (2002). The nonlinear knapsack problem—Algorithms and applications. *European Journal of Operational Research*, *138*(3), 459–472.

Chandra, A. K., Hirschberg, D. S., & Wong, C. K. (1976). Approximate algorithms for some generalized knapsack problems. *Theoretical Computer Science*, *3*(3), 293–304.

Chau, C., Elbassioni, K. M., & Khonji, M. (2016). Truthful mechanisms for combinatorial allocation of electric power in alternating current electric systems for smart grid. *ACM Transactions on Economics and Computation*, *5*(1), 7:1–7:29.

Chau, S. C., Elbassioni, K. M., & Khonji, M. (2014). Truthful mechanisms for combinatorial AC electric power allocation. In *International conference on autonomous agents and multi-agent systems, AAMAS '14* (pp. 1005–1012), Paris, France, 5–9 May 2014.

Csirik, J., Frenk, J. B. G., Labbé, M., & Zhang, S. (1991). Heuristic for the 0–1 min-knapsack problem. *Acta Cybernetica*, *10*(1–2), 15–20.

Elbassioni, K. M., & Nguyen, T. T. (2017). Approximation algorithms for binary packing problems with quadratic constraints of low cp-rank decompositions. *Discrete Applied Mathematics*, *230*, 56–70.

Garey, M., & Johnson, D. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. San Francisco: W.H. Freeman.

Hedengren, J. D. (2014). APMonitor Modeling Language. http://APMonitor.com. Accessed 18 Aug 2017.
Ibaraki, T., & Katoh, N. (1988). *Resource allocation problems*. Cambridge, MA: MIT Press.
Karapetyan, A., Khonji, M., Chau, C. K., Elbassioni, K., & Zeineldin, H. (2018). Efficient algorithm for scalable event-based demand response management in microgrids. *IEEE Transactions on Smart Grid*, *9*(4), 2714–2725. https://doi.org/10.1109/TSG.2016.2616945.
Kellerer, H., Pferschy, U., & Pisinger, D. (2004). *Knapsack problems*. Berlin: Springer.
Kellerer, H., & Strusevich, V. A. (2010). Fully polynomial approximation schemes for a symmetric quadratic knapsack problem and its scheduling applications. *Algorithmica*, *57*(4), 769–795.
Kellerer, H., & Strusevich, V. A. (2012). The symmetric quadratic knapsack problem: approximation and scheduling applications. *4OR*, *10*(2), 111–161.
Khonji, M., Karapetyan, A., Elbassioni, K., & Chau, C. K. (2016). Complex-demand scheduling problem with application in smart grid. In *Computing and combinatorics* (pp. 496–509). Berlin: Springer.
Nemhauser, G. L., & Wolsey, L. A. (1999). *Integer and combinatorial optimization*. New York, NY: Wiley-Interscience.
Pferschy, U., & Schauer, J. (2013). Approximating the quadratic knapsack problem on special graph classes. In *Approximation and online algorithms—11th international workshop, WAOA 2013* (pp. 61–72), Sophia Antipolis, France, September 5–6, 2013, Revised selected papers.
Renegar, J. (1992). On the computational complexity of approximating solutions for real algebraic formulae. *SIAM Journal on Computing*, *21*(6), 1008–1025.
Schrijver, A. (1986). *Theory of linear and integer programming*. New York: Wiley.
Rader, D. J, Jr., & Woeginger, G. J. (2002). The quadratic 0–1 knapsack problem with series-parallel support. *Operations Research Letters*, *30*(3), 159–166.
Woeginger, G. J. (2000). When does a dynamic programming formulation guarantee the existence of a fully polynomial time approximation scheme (FPTAS)? *INFORMS Journal on Computing*, *12*(1), 57–74.
Wood, A. J., & Wollenberg, B. F. (2012). *Power generation, operation, and control*. London: Wiley.
Yu, L., & Chau, C. (2013). Complex-demand knapsack problems and incentives in AC power systems. In *International conference on autonomous agents and multi-agent systems, AAMAS '13* (pp. 973–980), Saint Paul, MN, USA, May 6–10, 2013.