Mathematical Programming 57 (1992) 279-311
North-Holland

Approximation algorithms for indefinite quadratic programming

Stephen A. Vavasis
Department of Computer Science, Cornell University, Ithaca, NY, USA

Received 4 September 1991
Revised manuscript received 15 March 1992

We consider ε-approximation schemes for indefinite quadratic programming. We argue that such an approximation can be found in polynomial time for fixed ε and t, where t denotes the number of negative eigenvalues of the quadratic term. Our algorithm is polynomial in $1/\varepsilon$ for fixed t, and exponential in t for fixed ε.

We next look at the special case of knapsack problems, showing that a more efficient (polynomial in t) approximation algorithm exists.

1. Nonconvex quadratic programming

Quadratic programming is a nonlinear optimization problem of the following form:

$$\begin{aligned} &\text{minimize} \quad \tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}H\boldsymbol{x}+\boldsymbol{h}^{\mathrm{T}}\boldsymbol{x} \\ &\text{subject to} \quad W\boldsymbol{x}\geqslant \boldsymbol{b}. \end{aligned} \tag{1}$$

In this formulation, $\boldsymbol{x}$ is the n-vector of unknowns. The remaining variables stand for data in the problem instance: H is an $n\times n$ symmetric matrix, $\boldsymbol{h}$ is an n-vector, W is an $m\times n$ matrix, and $\boldsymbol{b}$ is an m-vector. The relation "$\geqslant$" in the constraint $W\boldsymbol{x}\geqslant\boldsymbol{b}$ is the usual componentwise inequality.

Quadratic programming, a generalization of linear programming, has applications in economics, planning, and many kinds of engineering design. In addition, more complicated kinds of nonlinear programming problems are often simplified into quadratic programming problems.

No efficient algorithm is known to solve the general case of (1). The lack of an efficient algorithm is not surprising, since quadratic programming is known to be NP-hard, a result due to Sahni (1974). More recently, Vavasis (1990) showed that the decision version of the problem lies in NP, and hence is NP-complete.

Correspondence to: Dr. Stephen A. Vavasis, Department of Computer Science, Cornell University, Ithaca, NY 14853, USA.

Part of this work was done while the author was visiting Sandia National Laboratories, Albuquerque, New Mexico, supported by the U.S. Department of Energy under contract DE-AC04-76DP00789. Part of this work was also supported by the Applied Mathematical Sciences Program (KC-04-02) of the Office of Energy Research of the U.S. Department of Energy under grant DE-FG02-86ER25013.A000 and in part by the National Science Foundation, the Air Force Office of Scientific Research, and the Office of Naval Research, through NSF grant DMS 8920550.

Many avenues for addressing (1) have been pursued in the literature. For example, efficient algorithms are known for the special case in which H is positive semidefinite, known as the *convex* case. See Kozlov, Tarasov and Hačijan (1979) for the first polynomial-time algorithm for the convex case. See Kapoor and Vaidya (1986) or Ye and Tse (1989) for efficient interior point algorithms for this problem. Active set methods (see Gill, Murray and Wright, 1981), a commonly-used class of methods for (1), are a combination of local search and heuristics.

A very successful way to address NP-hard combinatorial optimization problems has been approximation algorithms. In this report, we investigate ε-approximation algorithms for quadratic programming. In previous work (Vavasis, 1992a) we considered the *concave* case, that is, the case that H is negative semidefinite. The concave case, like the general case, is NP-hard.

First it is necessary to give a definition of ε-approximation:

Definition 1. Consider an instance of quadratic programming written in the form (1). Let $f(\boldsymbol{x})$ denote the objective function $\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}H\boldsymbol{x}+\boldsymbol{h}^{\mathrm{T}}\boldsymbol{x}$. Let $\boldsymbol{x}^*$ be an optimum point of the problem. We say that $\boldsymbol{x}^{\diamond}$ is an ε-approximate solution if there exists another feasible point $\boldsymbol{x}^{\#}$ such that

$$f(\boldsymbol{x}^{\diamond})-f(\boldsymbol{x}^*)\leqslant\varepsilon[f(\boldsymbol{x}^{\#})-f(\boldsymbol{x}^*)].$$

Notice that we may as well take $\boldsymbol{x}^{\#}$ in Definition 1 to be the point where the objective function is maximized. Thus, another way to interpret this definition is as follows. Let P denote the feasible region, and let interval $[a, b]$ be $f(P)$. Then $f(\boldsymbol{x}^{\diamond})$ should lie in the interval $[a, a+\varepsilon(b-a)]$.

Observe that any feasible point is a 1-approximation by this definition, and only the optimum is a 0-approximation. Thus, the definition makes sense only for ε in the interval $(0, 1)$. Our definition of approximation has been used in our earlier work, and also appears in other places such as Nemirovsky and Yudin (1983).

This definition has some useful properties. First, it is insensitive to translations or dilations of the objective function. In other words, if the objective function $f(\boldsymbol{x})$ is replaced by a new function $g(\boldsymbol{x})=af(\boldsymbol{x})+b$ where $a>0$, a vector $\boldsymbol{x}^{\diamond}$ that was previously an ε-approximation will continue to have that property. A second useful property is that ε-approximation is preserved under affine linear transformations of the feasible region.

We now state the first main theorem of this paper.

Theorem 2. *Consider the indefinite case of* (1). *Assume that the feasible region* $\{\boldsymbol{x}\colon W\boldsymbol{x}\geqslant\boldsymbol{b}\}$ *is compact. Let t be the number of negative eigenvalues of H. There is an algorithm to find an ε-approximate solution to* (1) *in*

$$\mathrm{O}(\lceil n(n+1)/\sqrt{\varepsilon}\rceil^t l)$$

steps. In this formula, l denotes the time to solve a convex quadratic programming problem of the same size as (1).

We remark that l grows polynomially with the size of the input, as mentioned above. The best asymptotic bound known for l in the special case of linear programming is due to Vaidya (1989). The assumption that the feasible region is compact is discussed further in the next section.

The algorithm we propose is based on covering the feasible region with small sets, and then enumerating those sets. Many covering algorithms have appeared in the literature. In particular, we refer the reader to Pardalos and Rosen (1987).

In Section 2 we provide the algorithm and prove Theorem 2. In Section 3 and Section 4 we describe a more efficient approximation algorithm for concave knapsack problems. This is extended to indefinite knapsack problems in Section 5. The knapsack problem has received attention in the literature because it is a simple special case of quadratic programming that exhibits many features of the general case. In Section 6 we indicate why polynomial dependence on $1/\varepsilon$ and exponential dependence on t in Theorem 2 is expected. In Section 7 we discuss open questions raised by this work.

The definition of approximation used for combinatorial optimization differs from out definition and is usually stated as follows. A feasible point $x^\diamond$ is an ε-approximation if

$$|f(x^\diamond)-f(x^*)|\leq \varepsilon \cdot f(x^*).$$

See, for example, Papadimitriou and Steiglitz (1982) for an extensive discussion of approximation for combinatorial optimization. This definition does not work for nonlinear optimization because it is not preserved when a constant is added to the objective function. In particular, the definition because useless in the case that $f(x^*)\leq 0$.

2. Proof of Theorem 2

The first part of the proof is a sequence of basis changes. First, we test whether the constraint set $\{x\colon Wx\geq b\}$ is full dimensional. This can be done by solving a single linear programming problem as shown by Freund, Roundy and Todd (1985). If not, a linear change of basis lowers the dimension of the problem and ensures without loss of generality that the feasible set is full dimensional. Note that this change of basis does not increase t, the number of negative eigenvalues of H (but it may cause t to decrease).

Let P denote the constraint set $P=\{x\colon Wx\geq b\}$. For the rest of this section, we assume that set P is compact. The difficulty if P is not compact is that we are not able to easily determine whether the original problem is unbounded. Indeed, Murty and Kabadi (1987) showed that it is NP-hard to determine whether an indefinite quadratic problem is unbounded. Let us state that as an assumption:

Assumption. We assume P is compact.

The next step is to compute a *weak Löwner–John* pair for set P. Recall that a *Löwner–John* pair for a convex body $P \subset \mathbb{R}^n$ is a pair of concentric ellipsoids E_1, E_2 such that $E_1 \subset P \subset E_2$ and E_1 is obtained from E_2 by shrinking each dimension by $1/n$. Such a pair always exists. A weak Löwner–John pair is defined analogously, except that the shrinking factor is $1/((n+1)\sqrt{n})$. Lovász (1986) shows how to compute a weak Löwner–John pair for a convex body in polynomial time.

Let us assume that the interior and exterior ellipsoids are defined by

$$E_1 = \{x \in \mathbb{R}^n\colon (x-c)^{\mathrm{T}} M (x-c) \leq 1\},$$
$$E_2 = \{x \in \mathbb{R}^n\colon (x-c)^{\mathrm{T}} M (x-c) \leq (n+1)^2 n\},$$

where M is a symmetric positive definite matrix, and c is some n-vector.

The next change of basis is to translate x by c, thereby centering the Löwner–John pair at the origin. This does not affect the quadratic term of the objective function. Next, find a nonsingular $n \times n$ matrix X such that $X^{\mathrm{T}}MX = I$ (where I denotes the identity) and such that $X^{\mathrm{T}}HX$ is diagonal. Such a matrix exists because M is positive definite, and it can be computed using standard eigenvalue methods. (See Golub and Van Loan (1989).)

Then change the basis again, replacing x by $X^{-1}x$. After this transformation, we can now make the following assumptions:

Assumption. The objective function has the form

$$\tfrac{1}{2}x^{\mathrm{T}}Dx + h^{\mathrm{T}}x$$

where D is diagonal.

Assumption. The constraint set, which we will continue to write as $P = \{x\colon Wx \geq b\}$, satisfies the containments $S_1 \subset P \subset S_2$ where

$$S_1 = \{x\colon x^{\mathrm{T}}x \leq 1\} \quad \text{and} \quad S_2 = \{x\colon x^{\mathrm{T}}x \leq n(n+1)^2\}.$$

The basis transformation by X does not change the signature of H, so we can assume that D has t negative diagonal entries. Let us split the vector x into two subvectors, $y \in \mathbb{R}^t$ and $z \in \mathbb{R}^{n-t}$, such that y corresponds to the negative entries of D. Then we can rewrite the problem as

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2}y^{\mathrm{T}}Ky + k^{\mathrm{T}}y + \tfrac{1}{2}z^{\mathrm{T}}Lz + l^{\mathrm{T}}z \\ &\text{subject to} && Ay + Bz \geq b \end{aligned} \tag{2}$$

where K is negative definite diagonal and L is positive semidefinite, and (A, B) represents a partitioning of the columns of W. Now, let $\hat{P}$ be the projection of P onto y-space, that is

$$\hat{P} = \{y\colon Ay + Bz \geqslant b \text{ for some } z \in \mathbb{R}^{n-t}\}.$$

Let $\hat{S}_1, \hat{S}_2$ be the projections of S_1, S_2 in y-space, that is

$$\hat{S}_1 = \{y\colon y^{\mathrm{T}} y \leqslant 1\} \quad \text{and} \quad \hat{S}_2 = \{y\colon y^{\mathrm{T}} y \leqslant n(n+1)^2\}.$$

Clearly $\hat{S}_1 \subset \hat{P} \subset \hat{S}_2$ since projection preserves containment.

Now we define the "projection" of the convex part of the objective function for $y \in \hat{P}$:

$$\phi(y) = \min\{\tfrac{1}{2} z^{\mathrm{T}} L z + l^{\mathrm{T}} z\colon z \in \mathbb{R}^{n-t},\ Ay + Bz \geqslant b\}.$$

With this definition of ϕ, the original problem can now be expressed simply as

$$\begin{aligned} &\text{minimize} && q(y) + \phi(y) \\ &\text{subject to} && y \in \hat{P} \end{aligned} \tag{3}$$

where

$$q(y) = \tfrac{1}{2} y^{\mathrm{T}} K y + k^{\mathrm{T}} y.$$

This is equivalent to (2). In particular, for any y_1 feasible for (2), $q(y_1) + \phi(y_1)$ will be the value of the minimum possible objective function value among all feasible vectors of the form (y_1, z).

We let $R \subset \mathbb{R}^t$ be the smallest rectangle containing $\hat{S}_2$, that is,

$$R = \underbrace{[-(n+1)\sqrt{n}, (n+1)\sqrt{n}] \times \cdots \times [-(n+1)\sqrt{n}, (n+1)\sqrt{n}]}_{t \text{ times}}.$$

The next step of the algorithm is to divide R into m^t subcubes. The integer m will be a number on the order of $n^2/\sqrt{\varepsilon}$; the exact formula is given below. Note that each subcube has side length $2(n+1)\sqrt{n}/m$.

Let the subcubes be denoted $R_1, \ldots, R_p$, where $p = m^t$. An example of the sets $\hat{P}$, $\hat{S}_1$, $\hat{S}_2$, R and $R_1, \ldots, R_p$ in the case $t = 2$, $m = 10$ is illustrated in Figure 1.

We next compute a linear approximation to the function $q(y)$ on each subcube. Let those linear approximations be denoted by $\lambda_1, \ldots, \lambda_p$. These linear approximations are interpolated from the vertices. In particular, focusing on a particular subcube R_i, which may be written as (say)

$$[a_1, b_1] \times \cdots \times [a_t, b_t],$$

we define

$$\lambda_i(y_1, \ldots, y_t) = \sum_{j=1}^{t} [\tfrac{1}{2} K_{jj}(a_j + b_j) y_j + c_j y_j - \tfrac{1}{2} k_{jj} a_j b_j]$$

where K_{jj}, k_j, y_j denote the components of K, $\boldsymbol{k}$, $\boldsymbol{y}$ respectively.

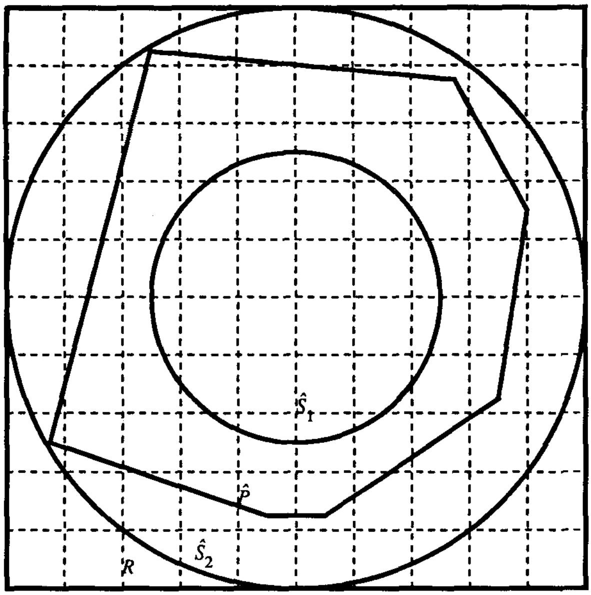

Fig. 1. Sets used by the approximation algorithm.

Next, we minimize $\lambda_i(y)+\phi(y)$ on R_i for each $i=1,\ldots,p$. This is equivalent to the following convex quadratic programming problem:

$$\begin{aligned} \text{minimize} \quad & \lambda_i(y)+\tfrac{1}{2}z^{\mathrm{T}}Lz+l^{\mathrm{T}}z \\ \text{subject to} \quad & Ay+Bz\geqslant b, \\ & y\in R_i. \end{aligned}$$

Let $(y^{\diamond}, z^{\diamond})$ be the pair with the minimum objective function value taken over all p convex quadratic programs of this form.

Let θ be the maximum absolute difference between λ_i and q in R_i. One easily checks (see, for example, Vavasis, 1992a) that

$$\theta=\left(\sum_{j=1}^{t}|K_{jj}|\right)\cdot\frac{(n+1)^2 n}{2m^2}. \tag{4}$$

Note that the factor on the right is one eighth of the square of the width of a subcube. We claim that $q(y^{\diamond})+\phi(y^{\diamond})$ is at most θ greater than optimal. This follows because $y^{\diamond}$ is a minimizer of the function $\lambda(y)+\phi(y)$. Here, λ is defined piecewise to agree with λ_i on R_i. The pointwise difference between $q+\phi$ and $\lambda+\phi$ is at most θ. Therefore, the difference between their minima is also at most θ.

Now, the next step is to show the existence of two points whose objective function values differ by at least θ/ε.

We first prove the following lemma about quadratic functions of one variable before addressing the general case. This lemma is also used later on in the discussion of knapsack problems.

Lemma 3. *Let $q_0(x)=ax^2+bx+c$ be a quadratic function defined on interval $[z_1, z_2]\subset\mathbb{R}$. Let l and u be the minimum and maximum values attained by q_0 on this interval. Then*

$$u-l\geqslant\tfrac{1}{4}|a|\cdot(z_2-z_1)^2.$$

Proof. We assume without loss of generality that $a\geqslant 0$, since the claim made in the lemma is preserved if $-q_0$ is substituted for q_0. Consider the point $z'=\frac{1}{2}(z_1+z_2)$. Then some algebra shows that

$$q_0(z_1)+q_0(z_2)-2q_0(z')=\tfrac{1}{2}a(z_2-z_1)^2.$$

Next, suppose that $q_0(z_1)\geqslant q_0(z_2)$ (the opposite case is similar). Then the previous inequality implies that

$$2q_0(z_1)-2q_0(z')\geqslant\tfrac{1}{2}a(z_2-z_1)^2.$$

Now the lemma follows, since $u\geqslant q_0(z_1)$ and $l\leqslant q_0(z')$. □

Now we return to the problem at hand. Let μ be the index such that $|K_{\mu\mu}|$ is maximized. Thus, we can estimate from (4) that

$$\theta\leqslant t|K_{\mu\mu}|\cdot\frac{(n+1)^2n}{2m^2}. \tag{5}$$

Consider the vector parameterized by α,

$$y(\alpha)=(0,\ldots,0,\alpha,0,\ldots,0),$$

where α occurs as the μth entry.

Note that $(y(\alpha),\mathbf{0})$ lies in S_1 for all $\alpha\in[-1,1]$, and hence in P. If we define $f(\alpha)$ to be the objective function in (2) evaluated at $(y(\alpha),\mathbf{0})$, then we see that f is a quadratic function of one variable whose leading coefficient is $\frac{1}{2}K_{\mu\mu}$. Therefore, by Lemma 3, f varies by at least $\frac{1}{2}|K_{\mu\mu}|$. This quantity is therefore a lower bound on the range of values of the objective function.

Thus, if we want to be assured that $(y^\diamond, z^\diamond)$ is an ε-approximate minimum, then it suffices to show that

$$\theta\leqslant\tfrac{1}{2}\varepsilon|K_{\mu\mu}|.$$

From (5), it suffices to establish the inequality

$$t|K_{\mu\mu}|\cdot\frac{(n+1)}{2m^2}\leqslant\tfrac{1}{2}\varepsilon|K_{\mu\mu}|.$$

The preceding inequality is solved by taking

$$m=\lceil(n+1)\sqrt{nt}/\sqrt{\varepsilon}\rceil.$$

This proves Theorem 2. □

In practice, a more efficient version of the algorithm proposed in this section would construct a hierarchy of subdivisions of R. For example, suppose that m is divisible by 2. Then we could partition R into a mesh with $\frac{1}{2}m$ grid cells in each dimension and carry out the algorithm of this section. The point returned by the algorithm from the coarse grid would not necessarily be an ε-approximate optimum, but the results would give good upper and lower bounds on the value of the objective function in each cube. We could then use these bounds to determine which grid cells need to be further subdivided. Applying these ideas in a hierarchical fashion would lead to a branch-and-bound style algorithm. See Pardalos and Rosen (1987) for algorithms for this style; see Papadimitriou and Steiglitz (1982) for a general description of branch and bound.

There are also possible improvements to the analysis of the algorithm. For example, instead of enumerating every rectangle R_i in R, we could limit attention to the rectangles that are contained or partly contained in $\hat{S}_2$. This would yield a better running time estimate, but still exponential in t.

3. The quadratic knapsack problem: Exact solution

In this section and the next two sections we show that a more efficient algorithm is possible for the quadratic knapsack problem. "More efficient" means polynomial in t and $1/\varepsilon$. The algorithm we propose has the disadvantage that it is only weakly polynomial in the problem size (i.e., the number of arithmetic operations depends on the number of bits in the problem). This is in contrast to other algorithms for knapsack problems mentioned below.

The quadratic knapsack problem has the following form:

$$\begin{aligned} &\text{minimize} && \boldsymbol{x}^{\mathrm{T}}D\boldsymbol{x}+\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \\ &\text{subject to} && \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}=\gamma, \\ & && l_i \leqslant x_i \leqslant u_i, \quad i=1,\ldots,n. \end{aligned} \tag{6}$$

Here, D is a diagonal matrix. Thus, the objective function is *separable*, that is, it can be written in the form

$$q_1(x_1)+\cdots+q_n(x_n)$$

where each q_i is a quadratic function of one variable. It will be necessary below to make an explicit reference to the coefficients of q_i. Accordingly, we write those coefficients as

$$q_i(x)=d_ix^2+c_ix.$$

In other words, d_i is the ith diagonal entry of matrix D, and c_i is the ith component of vector $\boldsymbol{c}$.

The quadratic knapsack problem arises in resource allocation applications. It also arises as a subproblem in algorithms for more general optimization problems. See

Cottle, Duvall and Zikan (1986) for an application and further references. The convex case of this problem, that is, the case in which all the diagonal entries of D are nonnegative, can be solved in $O(n \log n)$ as shown by Helgason, Kennington and Lall (1980). This bound was improved to $O(n)$ by Brucker (1984). The nonconvex case is NP-hard, as proved by Sahni (1974). Polynomial-time algorithms for finding local minima for nonconvex cases have been given by Moré and Vavasis (1991) and Vavasis (1992b).

In this section we propose an approximation algorithm for solving (6) based on *dynamic programming* (see Bellman, 1957). The observation that knapsack-type problems are amenable to approximation is not novel: see Papadimitriou and Steiglitz (1982) for a dynamic programming approach to a combinatorial knapsack problem. Our contribution is to show that a particular dynamic programming approach is an efficient approximation algorithm for the sense of approximation proposed in Definition 1.

We start by focusing on the concave case of (6), that is, the case that all the diagonal entries of D are nonpositive. In Section 5 we show how to extend the algorithm to the fully indefinite case.

Before describing the approximation algorithm, we first give an exact dynamic programming algorithm for the concave case of (6). This algorithm is exponential time, but will serve as the basis for a polynomial-time approximation algorithm.

First, we start with an assumption to simplify the notation: we assume that the vector $\boldsymbol{a}$ in (6) is actually the vector of all 1's, so that (6) may be rewritten:

$$\begin{aligned} \text{minimize} \quad & q_1(x_1)+\cdots+q_n(x_n) \\ \text{subject to} \quad & x_1+\cdots+x_n=\gamma, \\ & l_i \leq x_i \leq u_i, \quad i=1,\ldots,n. \end{aligned} \tag{7}$$

This assumption is made without loss of generality because we can always scale variable x_i by $1/a_i$ to put the problem in this form. (Note that if $a_i=0$, then x_i is decoupled from the rest of the problem and may be deleted and optimized separately. Note also that scaling x_i by $1/a_i$ does not affect the sign of the corresponding diagonal entry in D regardless of the sign of a_i, so that this transformation does not change the concavity or convexity of the problem.)

We also assume that $l_i<u_i$ for $i=1,\ldots,n$, since if $l_i=u_i$ for some i, then x_i is uniquely determined and may be deleted. If $l_i>u_i$, then (7) is infeasible.

The exact algorithm is based on constructing a sequence of real-valued functions $\mu^{(k)}$ of one variable. The domain of $\mu^{(k)}$ is the interval $I_k=[l_1+\cdots+l_k, u_1+\cdots+u_k]$. Let the endpoints of I_k be denoted as λ_k and ω_k. The actual definition of $\mu^{(k)}(\zeta)$ for $\zeta \in I_k$ is as follows:

$$\begin{aligned} \mu^{(k)}(\zeta) \quad = \quad \text{minimum of} \quad & q_1(x_1)+\cdots+q_k(x_k) \\ \text{subject to} \quad & x_1+\cdots+x_k=\zeta, \\ & l_i \leq x_i \leq u_i, \quad i=1,\ldots,k. \end{aligned} \tag{8}$$

Notice that the optimal solution to (7) is exactly $\mu^{(n)}(\gamma)$. Thus, if we had an exact representation of function $\mu^{(n)}$ available, (7) would be solved.

The algorithm constructs an exact representation of $\mu^{(k+1)}$ inductively from $\mu^{(k)}$. Clearly it is easy to compute $\mu^{(1)}$: function $\mu^{(1)}(\zeta)$ is a concave quadratic function of one variable (equal to function q_1). The next theorem shows how it is possible to obtain $\mu^{(k+1)}$.

Theorem 4. *Each function $\mu^{(k)}$ is a continuous piecewise quadratic function with a finite number of breakpoints; the pieces are always concave. Moreover, for $k>1$, $\mu^{(k)}(\zeta)$ is equal to either:*

(i) *$\mu^{(k-1)}(b)+q_k(\zeta-b)$, where b is some breakpoint of $\mu^{(k-1)}$, or*

(ii) *$\mu^{(k-1)}(\zeta-t)+q_k(t)$, where t is either l_k or u_k.*

Remarks. 1. We consider the endpoints of I_{k-1} (namely, $\lambda_{k-1}=l_1+\cdots+l_{k-1}$ and $\omega_{k-1}=u_1+\cdots+u_{k-1}$) as breakpoints of $\mu^{(k-1)}$ for case (i) in the theorem.

2. An example of a function $\mu^{(n)}$ is plotted in Figure 2. This example was generated from a random instance of (8) with $n=6$.

Proof of Theorem 4 (*Sketch*). First, notice that $\mu^{(k)}$ can be entirely determined from $\mu^{(k-1)}$ because of the "principal of optimality" common to many dynamic programming approaches. We omit the argument, but provide the recursive formula

$$\mu^{(k)}(\zeta)=\min\{\mu^{(k-1)}(\zeta-t)+q_k(t):\ t\in[l_k,u_k],\ \zeta-t\in[\lambda_{k-1},\omega_{k-1}]\}. \tag{9}$$

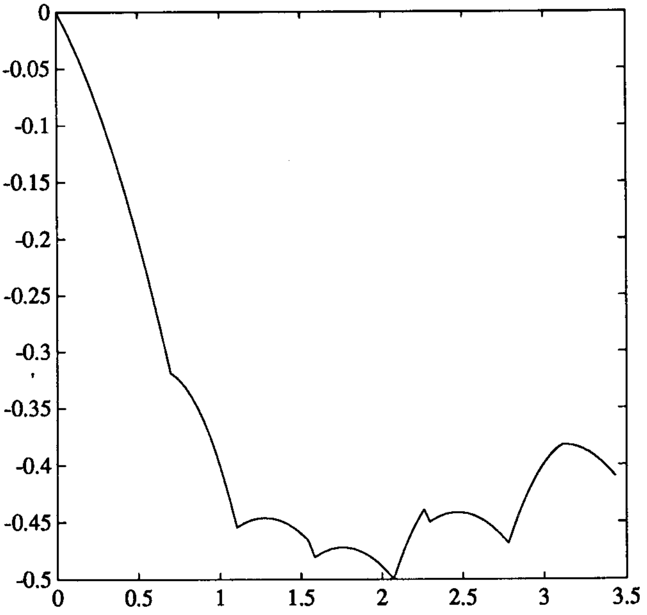

Fig. 2. An example of $\mu^{(n)}$.

Once this equation is established, everything stated in the theorem follows from the well-known fact that the minimizer of a concave function defined on an interval is always achieved at one of the endpoints of the interval. The full proof of this theorem appears in Vavasis (1991). □

The preceding theorem suggests an algorithm for finding an explicit representation for $\mu^{(k)}$ from $\mu^{(k-1)}$. Specifically, write down the finite list of piecewise quadratic functions of ζ defined either by the formula in case (i) or in case (ii). These are functions of ζ defined on subintervals of I_k. Then compute their pointwise minimum by scanning the pieces from left to right and computing intersections. We call the algorithm for constructing $\mu^{(k)}$ from $\mu^{(k-1)}$ *Scan-Breakpoint.* We call the entire minimization algorithm *Exact DP* (here, DP stands for dynamic programming). We discuss efficient implementations of Scan-Breakpoint in the next section.

Exact DP is exponential-time in the worst case because the number of breakpoints in $\mu^{(k)}$ can grow exponentially with k. (It should not be a surprise that the algorithm is not polynomial-time — minimizing (7) in the concave case is NP-hard.)

Below we propose an approximation algorithm called *Approx DP* based on Exact DP. The basic idea of Approx DP is straightforward: instead of computing the sequence of functions $\mu^{(1)}, \mu^{(2)}, \ldots$, we compute approximations: $\pi^{(1)}, \pi^{(2)}, \ldots$.

The main idea underlying the approximation scheme is that $\mu^{(k)}$ on a sufficiently small interval may be replaced by a linear interpolation without introducing too much error. The reason for the interpolation is to bound the number of breakpoints that occur in $\pi^{(k)}$. Because of the way "approximation" is defined by Definition 1, the bound on the error from linear interpolation must have the form that the difference between $\mu^{(k)}(\zeta)$ and a linear approximation to it is bounded above by a (small) constant multiplied by the range of the objective function in (8).

Providing such a bound is the purpose of the upcoming theorem, the main result of this section. In order to state the theorem, we need a function to describe the objective function range in (8). Let

$$\begin{aligned} \rho^{(k)}(\zeta) \quad = \quad & \text{maximum of} \quad q_1(x_1)+\cdots+q_k(x_k) \\ & \text{subject to} \quad x_1+\cdots+x_k=\zeta, \\ & \qquad\qquad\quad l_i \leq x_i \leq u_i, \quad i=1,\ldots,k. \end{aligned} \tag{10}$$

Thus, the definition of $\rho^{(k)}$ is identical to the definition of $\mu^{(k)}$ except we take a maximum instead of a minimum. We remark that $\rho^{(k)}$ has considerably more structure than $\mu^{(k)}$; it is globally concave, has a linear number of breakpoints, and can be computed in polynomial time. We return to this subject later.

The main theorem about approximation with linear interpolation requires two more lemmas about quadratic functions of one variable. The first lemma says that if q is an initially increasing concave quadratic function that stays positive, then it must have a certain range.

Lemma 5. *Let $q(x)$ be a concave quadratic function such that $q(0)=0$, $q'(0)=b$ with $b \geqslant 0$. Suppose $r \geqslant 0$ and suppose that q is nonnegative on $[0, r]$. Then the maximum value of q on $[0, r]$ is at least $\frac{1}{4}br$.*

Proof. We can write $q(x)=ax^2+bx$, where a is some nonpositive real number and b is as in the lemma. If $a=0$, then the result is obvious since $q(r)=br$. Similarly, if $r=0$ or $b=0$ then the result is immediate. Thus, assume that $a<0$, $r>0$, $b>0$. The fact that q is nonzero on $[0, r]$ means that $ar^2+br \geqslant 0$. Since q is concave, its maximum value overall is attained at $-b/(2a)$.

On interval $[0, r]$ the maximum is attained either at $-b/(2a)$ or at r, whichever is smaller. Suppose the maximum is attained at r so that $r \leqslant -b/(2a)$, i.e., $a \geqslant -b/(2r)$. Then we compute that

$$q(r)=ar^2+br \geqslant -b/(2r) \cdot r^2+br \geqslant \tfrac{1}{2}br.$$

Thus, the lemma is proved in this case.

If the maximum is attained at $-b/(2a)$, then we compute that

$$q(-b/(2a))=a \cdot (-b/(2a))^2+b \cdot (-b/(2a))=-b^2/(4a).$$

Now we use the fact that $ar^2+br \geqslant 0$, i.e., $a \geqslant -b/r$ to obtain $q(-b/(2a)) \geqslant \frac{1}{4}br$. □

The next lemma gives a lower bound on the leading coefficient of a concave quadratic function in the case that its rate of increase drops by a factor of $\frac{1}{2}$ or more.

Lemma 6. *Let $q(x)$ be a concave quadratic function of one variable, and v, w two real numbers such that $0<v<w$. Let p denote $q(v)-q(0)$, and suppose that $p>0$, and suppose also that*

$$q(w)-q(0) \leqslant \frac{w}{2v} \cdot p.$$

Then the leading coefficient of q is at least $p/[2v(w-v)]$ in absolute value.

Proof. Let $q(x)=ax^2+bx+c$, with $a \leqslant 0$ by assumption. Then the inequality in the lemma becomes

$$aw^2+bw \leqslant \frac{w}{2v}(av^2+bv)$$

i.e.,

$$aw^2+\tfrac{1}{2}bw-\tfrac{1}{2}avw \leqslant 0.$$

Now, $p=av^2+bv$, so $b=p/v-av$, thus we have

$$aw^2+\tfrac{1}{2}(p/v-av)w-\tfrac{1}{2}avw \leqslant 0.$$

Simplifying and solving for a gives the result. □

Now we state the key theorem about approximating $\mu^{(k)}$. Inequality (13) below bounds the difference between $\mu^{(k)}$ and a linear approximation to this function in terms of the range of the objective function.

Theorem 7. *Let b_1, b_2 be two points in I_k such that $b_1 < b_2$. Suppose $\zeta \in [b_1, b_2]$; in particular, write*

$$\zeta = (1-\phi)b_1 + \phi b_2$$

for $\phi \in [0, 1]$. Let us also make the following assumptions.

$$\text{(i)} \quad b_2 - b_1 \leq \frac{\min(\omega_k - b_1, b_2 - \lambda_k)}{24k}, \tag{11}$$

and

(ii) *For each $j = 1, \ldots, k$,*

$$u_j - l_j \leq \tfrac{1}{2} \sum_{i=1}^{k} (u_i - l_i) \tag{12}$$

(*in particular, $k > 1$*).

Let h denote the value of $\mu^{(k)}$ interpolated linearly from b_1, b_2 evaluated at ζ, i.e.,

$$h = (1-\phi)\mu^{(k)}(b_1) + \phi\mu^{(k)}(b_2).$$

Then

$$|\mu^{(k)}(\zeta) - h| \leq y \cdot (\rho^{(k)}(\zeta) - \mu^{(k)}(\zeta)) \tag{13}$$

where

$$y = 72k \cdot \frac{b_2 - b_1}{\min(\omega_k - b_1, b_2 - \lambda_k)}. \tag{14}$$

Remarks. Before proving the theorem, we make a few remarks about its contents. Observe that the parenthesized expression on the right-hand side of (13) is the range of the objective function in (8). Notice also, that in order for (11) to hold, and for y to remain small in (14), the size of the interval $[b_1, b_2]$ must shrink as the interval approaches either endpoint of $[\lambda_k, \omega_k]$. In other words, the linear approximation must become finer near the endpoints of I_k. This is because range of the objective function shrinks at these points, so we would expect that the approximation must be more accurate in an absolute sense. In the extreme case, $\mu^{(k)}(\lambda_k) = \rho^{(k)}(\lambda_k)$ and similarly for ω_k.

The final remark concerns the condition that $u_j - l_j$ be no more than half $\omega_k - \lambda_k$. This condition makes the approximation process (described below) more complicated. Unfortunately, this condition is necessary because the theorem becomes false if it is dropped. (See Vavasis (1991) for a counterexample.)

Proof of Theorem 7. We introduce more notation. We assume k is fixed, and let f denote the objective function in (8) (i.e., $f(x) = q_1(x_1) + \cdots + q_k(x_k)$). Let v denote the optimizer when ζ is taken to be b_1 in (8). In other words, $f(\boldsymbol{v})=\mu^{(k)}(b_1)$, $v_1+\cdots+v_k=b_1$, and $l_i\leq v_i\leq u_i$ for $i=1,\ldots,k$. Let z be the optimal vector in (8) for ζ, and let $\boldsymbol{w}$ be the vector corresponding to $\mu^{(k)}(b_2)$.

The theorem defines h to be the linear interpolation of $f(\boldsymbol{v})$ and $f(\boldsymbol{w})$:

$$h=(1-\phi)f(\boldsymbol{v})+\phi f(\boldsymbol{w}).$$

Thus, the goal is to bound $|f(z)-h|$.

We consider several cases. In each case, we find two feasible points for (8) whose difference in objective function values is a multiple of $|f(z)-h|$. The proofs in each of the cases seem to be fairly technical; we have not identified a principle underlying this theorem.

Case 1, $f(z)\leq h$. We claim that there is at least one component, say z_σ, of z such that $u_\sigma-z_\sigma\geq(\omega_k-\zeta)/k$. This follows immediately when we notice that the sum of u_i-z_i for $i=1,\ldots,k$ is exactly $\omega_k-\zeta$. Similarly, there is at least one component, say z_τ, such that $z_\tau-l_\tau\geq(\zeta-\lambda_k)/k$.

If we relax the conditions for σ and τ, and require only that

$$u_\sigma-z_\sigma\geq\frac{\omega_k-\zeta}{3k}\tag{15}$$

and

$$z_\tau-l_\tau\geq\frac{\zeta-\lambda_k}{3k},\tag{16}$$

then we can assume without loss of generality that $\sigma\neq\tau$. The reason for this is as follows. Suppose that only one index σ satisfies (15). Then by summing up the values of u_i-z_i for $i\neq\sigma$, we can deduce that $u_\sigma-z_\sigma\geq\frac{2}{3}(\omega_k-\zeta)$. Similarly, supposing that only one index τ satisfies (16) leads to the conclusion that $z_\tau-l_\tau\geq\frac{2}{3}(\zeta-\lambda_k)$. Finally, assuming that $\sigma=\tau$, we add the two preceding inequalities to obtain

$$u_\sigma-l_\sigma\geq\tfrac{2}{3}(\omega_k-\lambda_k).$$

But this contradicts (12).

Thus, we assume $\sigma\neq\tau$. Next, we introduce the following real-valued quadratic function:

$$g(r)=f(z+r\boldsymbol{e}_\sigma-r\boldsymbol{e}_\tau)-f(z).\tag{17}$$

Here, $\boldsymbol{e}_\sigma$ is the vector in $\mathbb{R}^k$ whose entry in position σ is 1, and whose other entries are zeros. Vector $\boldsymbol{e}_\tau$ is defined similarly.

Note that the vector $z+r\boldsymbol{e}_\sigma-r\boldsymbol{e}_\tau$ is feasible for (8) for r between 0 and r_1, where

$$r_1=\min(u_\sigma-z_\sigma,\,z_\tau-l_\tau).$$

In particular, the components of $z+r\boldsymbol{e}_\sigma+r\boldsymbol{e}_\tau$ add up to ζ, and each component is between l_i and u_i.

Thus, $g(r)\geq0$ for all $r\in[0,r_1]$ because $f(z)$ is the minimizer in (8).

Next, observe that we can come up with an explicit formula for $g(r)$ using the notation introduced earlier for the coefficients of the q_i's,

$$g(r)=(c_\sigma+2d_\sigma z_\sigma-c_\tau-2d_\tau z_\tau)r+(d_\sigma+d_\tau)r^2. \tag{18}$$

We now obtain a lower bound on the linear coefficient in $g(r)$. Observe that

$$f(z+(b_2-\zeta)e_\sigma)-f(w)\geqslant 0 \tag{19}$$

because w is the minimizer in (8) for feasible vectors whose components add to b_2, and the components of $z+(b_2-\zeta)e_\sigma$ add to b_2. Moreover, each component of $z+(b_2-\zeta)e_\sigma$ lies between l_i and u_i. Clearly the only component that needs attention in this regard is the σ component, whose value is $z_\sigma+b_2-\zeta$. First, notice from (1) that

$$\begin{aligned}\frac{\omega_k-\zeta}{3k}&=\frac{\omega_k-b_1}{3k}-\frac{\zeta-b_1}{3k}\\ &\geqslant\frac{\omega_k-b_1}{3k}-\frac{b_2-b_1}{3k}\\ &\geqslant\frac{\omega_k-b_1}{3k}-\frac{\omega_k-b_1}{(3k)\cdot(24k)}\\ &\geqslant\frac{\omega_k-b_1}{3k}-\frac{\omega_k-b_1}{72k}\\ &\geqslant\frac{23(\omega_k-b_1)}{72k}.\end{aligned} \tag{20}$$

Now we analyze the σ component of $z+(b_2-\zeta)e_\sigma$. Once again we use (11) as well as (15) and (20).

$$\begin{aligned}z_\sigma+b_2-\zeta&\leqslant z_\sigma+b_2-b_1\\ &\leqslant u_\sigma-(u_\sigma-z_\sigma)+b_2-b_1\\ &\leqslant u_\sigma-\frac{\omega_k-\zeta}{3k}+b_2-b_1\\ &\leqslant u_\sigma-\frac{23(\omega_k-\zeta)}{72k}+\frac{\omega_k-\zeta}{24k}\\ &\leqslant u_\sigma.\end{aligned} \tag{21}$$

This proves (19). We can rewrite (19) as

$$(c_\sigma+2d_\sigma z_\sigma)(b_2-\zeta)+d_\sigma(b_2-\zeta)^2\geqslant f(w)-f(z).$$

By comparing $f(z-(\zeta-b_1)e_\tau)$ to $f(v)$ we obtain in a similar fashion the inequality

$$-(c_\tau+2d_\tau z_\tau)(\zeta-b_1)+d_\tau(\zeta-b_1)^2\geqslant f(v)-f(z).$$

If we divide the first inequality by $b_2-\zeta$ and the second by $\zeta-b_1$ and add them together, we obtain

$$c_\sigma+2d_\sigma z_\sigma-c_\tau-2d_\tau z_\tau+d_\sigma(b_2-\zeta)+d_\tau(\zeta-b_1)\geq\frac{f(w)-f(z)}{b_2-\zeta}+\frac{f(v)-f(z)}{\zeta-b_1}.$$

Now we observe that the last two terms on the left-hand side are both negative, so dropping them does not change the truthfulness of the inequality. On the right-hand side, we can take a common denominator, plug in the definition of ϕ, and rearrange to obtain

$$c_\sigma+2d_\sigma z_\sigma-c_\tau-2d_\tau z_\tau\geq\frac{(b_2-b_1)[\phi f(w)+(1-\phi)f(v)-f(z)]}{(b_2-\zeta)(\zeta-b_1)}.$$

Notice that the expression in square brackets is exactly $h-f(z)$, where h was defined above as the interpolation of $f(v)$ and $f(w)$. Thus, we have

$$c_\sigma+2d_\sigma z_\sigma-c_\tau-2d_\tau z_\tau\geq\frac{(b_2-b_1)(h-f(z))}{(b_2-\zeta)(\zeta-b_1)}.$$

Notice that the expression on the left-hand side of this inequality is the linear coefficient of $g(r)$.

Thus, we have a lower bound on the linear coefficient of $g(r)$ (nonnegative because the hypothesis for this case is that $h\geq f(z)$), and we also know that $g(r)$ is nonnegative on $[0, r_1]$ and zero at the origin. Therefore, we can apply Lemma 5 to conclude that g attains a value of at least

$$\frac{r_1(b_2-b_1)(h-f(z))}{4(b_2-\zeta)(\zeta-b_1)}.$$

Recall that $g(r)+f(z)$ corresponds to a feasible value for (8) on $[0, r_1]$. Thus,

$$\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)\geq\frac{r_1(b_2-b_1)\cdot|h-\mu^{(k)}(\zeta)|}{4(b_2-\zeta)(\zeta-b_1)}.$$

Now we notice that

$$\frac{b_2-b_1}{(b_2-\zeta)(\zeta-b_1)}$$

is at least $4/(b_2-b_1)$. Also, r_1 (defined above) is at least $\min(\omega_k-\zeta, \zeta-\lambda_k)/(3k)$ by choice of σ and τ. Thus,

$$\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)\geq\frac{\min(\omega_k-\zeta, \zeta-\lambda_k)}{3k(b_2-b_1)}\cdot|h-\mu^{(k)}(\zeta)|.$$

Now we apply (20) and its analog for $\zeta-\lambda_k$ to conclude that

$$\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)\geq\frac{23\min(\omega_k-b_1, b_2-\lambda_k)}{72k(b_2-b_1)}\cdot|h-\mu^{(k)}(\zeta)|.$$

This proves the theorem.

Case 2, $f(z) \geqslant h$. (This case will be subdivided into Cases 3, 4, 5 and 6.) Again, we start this case by identifying particular subscripts σ and τ. Define two index sets J and K as follows:

$$J = \left\{ i\colon u_i - v_i \geqslant \frac{\omega_k - b_1}{3k} \right\}, \qquad K = \left\{ i\colon w_i - l_i \geqslant \frac{b_2 - \lambda_k}{3k} \right\}.$$

Note that an averaging argument similar to the argument of Case 1 shows that J, K are both nonempty.

Case 3, $f(z) \geqslant h$, and there exists either a $\sigma \in J$ such that $w_\sigma - v_\sigma \geqslant \frac{1}{3}(u_\sigma - v_\sigma)$, or a $\sigma \in K$ such that $w_\sigma - v_\sigma \geqslant \frac{1}{3}(v_\sigma - l_\sigma)$. Note that in either choice for this subcase, we can conclude that

$$w_\sigma - v_\sigma \geqslant \frac{\min(\omega_k - b_1, b_2 - \lambda_k)}{9k} \tag{22}$$

by definition of J and K.

In this case, let

$$g_1 = (1-\phi) f(\boldsymbol{v} + (\zeta - b_1)\boldsymbol{e}_\sigma) + \phi f(\boldsymbol{w} - (b_2 - \zeta)\boldsymbol{e}_\sigma).$$

Notice that g_1 is the weighted average of f evaluated at two points (namely, $\boldsymbol{v} + (\zeta - b_1)\boldsymbol{e}_\sigma$ and $\boldsymbol{w} - (b_2 - \zeta)\boldsymbol{e}_\sigma$) feasible for (8). The fact that $\boldsymbol{w} - (b_2 - \zeta)\boldsymbol{e}_\sigma$ is feasible for (8) is proved as follows (the proof for $\boldsymbol{v} + (\zeta - b_1)\boldsymbol{e}_\sigma$ is similar). Clearly the components sum to ζ, and all components are between l_i and u_i except possibly the σ component. Now we have a calculation for this component analogous to (21) using (11) and (22):

$$\begin{aligned} w_\sigma + \zeta - b_2 &\geqslant l_\sigma + (w_\sigma - l_\sigma) + b_1 - b_2 \\ &\geqslant l_\sigma + (w_\sigma - v_\sigma) + b_1 - b_2 \\ &\geqslant l_\sigma + \frac{\min(\omega_k - b_1, b_2 - \lambda_k)}{9k} - \frac{\min(\omega_k - b_1, b_2 - \lambda_k)}{24k} \\ &\geqslant l_\sigma. \end{aligned}$$

Therefore, $g_1 \geqslant f(z)$. We can come up with an explicit expression for g_1 in terms of the objective function coefficients:

$$\begin{aligned} g_1 &= h + 2d_\sigma(b_2 - b_1)(v_\sigma - w_\sigma)\phi(1-\phi) + d_\sigma \phi(1-\phi)(b_2 - b_1)^2 \\ &\leqslant h + 2d_\sigma(b_2 - b_1)(v_\sigma - w_\sigma)\phi(1-\phi). \end{aligned}$$

Note that the last term of the right-hand side of the first line is negative and hence was dropped.

Now we define

$$g_2 = f((1-\phi)\boldsymbol{v} + \phi \boldsymbol{w}).$$

Once again, observe that the argument to f is feasible for (8). Therefore, $g_2 \leqslant \rho^{(k)}(\zeta)$. We can write an explicit expression for g_2 as follows:

$$g_2 = h - \phi(1-\phi) \sum_{i=1}^{k} (v_i - w_i)^2 d_i \geqslant h - \phi(1-\phi)(v_\sigma - w_\sigma)^2 d_\sigma.$$

Notice that all terms of the summation are negative.

Comparing the inequalities for g_1 and g_2, we conclude that

$$g_2 - h \geqslant \frac{w_\sigma - v_\sigma}{2(b_2 - b_1)}(g_1 - h).$$

Plugging in other inequalities so far including (22) gives

$$\rho^{(k)}(\zeta) - h \geqslant \frac{\min(\omega_k - b_1, b_2 - \lambda_k)}{(18k)(b_2 - b_1)}(\mu^{(k)}(\zeta) - h).$$

Subtract $\mu^{(k)}(\zeta) - h$ from both sides to obtain

$$\rho^{(k)}(\zeta) - \mu^{(k)}(\zeta) \geqslant \frac{\min(\omega_k - b_1, b_2 - \lambda_k) - (18k)(b_2 - b_1)}{(18k)(b_2 - b_1)}(\mu^{(k)}(\zeta) - h).$$

Using (11), the numerator can be simplified to obtain

$$\rho^{(k)}(\zeta) - \mu^{(k)}(\zeta) \geqslant \frac{\min(\omega_k - b_1, b_2 - \lambda_k)}{(72k)(b_2 - b_1)}(\mu^{(k)}(\zeta) - h).$$

Case 4, $f(z) \geqslant h$, and for all $i \in J$, $w_i - v_i \leqslant \frac{1}{3}(u_i - v_i)$ and for all $i \in K$, $w_i - v_i \leqslant \frac{1}{3}(v_i - l_i)$. We select a $\sigma \in J$ and a $\tau \in K$. We claim that without loss of generality, we may assume $\sigma \neq \tau$. Suppose to the contrary that $J = K = \{\sigma\}$; we will argue that this leads to a contradiction.

We note by definition of J that the sum of $u_i - v_i$ for $i \notin J$ is at most $\frac{1}{3}(\omega_k - b_1)$. On the other hand, the sum of $u_i - v_i$ overall is equal to $\omega_k - b_1$. Therefore, if $J = \{\sigma\}$ we conclude that

$$u_\sigma - v_\sigma \geqslant \tfrac{2}{3}(\omega_k - b_1).$$

Similar reasoning from the premise that $K = \{\sigma\}$ shows that

$$w_\sigma - l_\sigma \geqslant \tfrac{2}{3}(b_2 - \lambda_k).$$

Adding the two preceding inequalities shows that

$$u_\sigma - l_\sigma + w_\sigma - v_\sigma \geqslant \tfrac{2}{3}(\omega_k - \lambda_k). \tag{23}$$

Now we obtain an upper bound on $w_\sigma - v_\sigma$. Note that since $\sigma \in J \cap K$, by the hypothesis for this case,

$$w_\sigma - v_\sigma \leqslant \tfrac{1}{3}(u_\sigma - v_\sigma)$$

and

$$w_\sigma - v_\sigma \leqslant \tfrac{1}{3}(w_\sigma - l_\sigma).$$

Adding these two inequalities and simplifying yields

$$w_\sigma - v_\sigma \leq \tfrac{1}{5}(u_\sigma - l_\sigma).$$

Now we combine this inequality with (23) to obtain

$$\tfrac{6}{5}(u_\sigma - l_\sigma) \geq \tfrac{2}{3}(w_\kappa - \lambda_k).$$

But this contradicts (12). Thus, without loss of generality, $\sigma \neq \tau$.

Now, define two concave quadratic functions of one variable r as follows:

$$g_1(r) = (1-\phi)f\left(v + \frac{r}{1-\phi}e_\sigma\right) + \phi f\left(w - \frac{r}{\phi}e_\tau\right)$$

and

$$g_2(r) = f((1-\phi)v + \phi w + r(e_\sigma - e_\tau)).$$

We make several observations about these functions. First, observe that $g_2(r) \geq g_1(r)$ for all values of r because f is a concave function (g_1 corresponds to a weighted average of f at two points, g_2 corresponds to f evaluated at the weighted average of the points).

Second, note that $g_1(0) = h$; this follows from the definition of h. Define

$$r_1 = \phi(1-\phi)(b_2 - b_1).$$

Note that $0 \leq r_1 \leq \tfrac{1}{4}(b_2 - b_1)$. The third observation is that $g_1(r_1) \geq f(z)$. This is because $g_1(r_1)$ corresponds to a weighted average of f evaluated at two points feasible for (8), namely, $v + r_1 e_\sigma/(1-\phi)$ and $w - r_1 e_\tau/\phi$. In particular, the sums of the components at each of these points is equal to ζ. Also, each component of these two vectors is between l_i and u_i. For example, we can check on component σ of $v + r_1 e_\sigma/(1-\phi)$ which is $v_\sigma + (\zeta - b_1)$:

$$\begin{aligned} v_\sigma + (\zeta - b_1) &\leq u_\sigma - (u_\sigma - v_\sigma) + (b_2 - b_1) \\ &\leq u_\sigma - \frac{\omega_k - b_1}{3k} + \frac{\omega_k - b_1}{24k} \\ &\leq u_\sigma. \end{aligned}$$

Thus, the weighted average $g_1(r_1)$ must be greater than or equal to $f(z)$, which is the minimum for (8).

Next, define

$$r_2 = \min(u_\sigma - ((1-\phi)v_\sigma + \phi w_\sigma), (1-\phi)v_\tau + \phi w_\tau - l_\tau).$$

Observe that $(1-\phi)v + \phi w + r(e_\sigma - e_\tau)$ is feasible for (8) for all r between 0 and r_2 (i.e., the components add to ζ and are between l_i and u_i). Thus, $g_2(r) \geq f(z)$ for all

$r \in [0, r_2]$. Notice also that

$$\begin{aligned} u_\sigma - ((1-\phi)v_\sigma + \phi w_\sigma) &= u_\sigma - v_\sigma + \phi(v_\sigma - w_\sigma) \\ &\geqslant u_\sigma - v_\sigma - \tfrac{1}{3}\phi(u_\sigma - v_\sigma) \\ &\geqslant u_\sigma - v_\sigma - \tfrac{1}{3}(u_\sigma - v_\sigma) \\ &\geqslant \tfrac{2}{3}(u_\sigma - v_\sigma) \\ &\geqslant 2(\omega_k - b_1)/(9k). \end{aligned}$$

This puts a lower bound on the first term in the definition of r_2. A lower bound in terms of $b_2 - \lambda_k$ is similarly obtained for the second term. Note that in the above chain of inequalities, we used the hypothesis for Case 4, the definition of J, and the fact that $\phi \leqslant 1$.

Thus, we conclude that

$$r_2 \geqslant 2\min(\omega_k - b_1, b_2 - \lambda_k)/(9k).$$

Let $p = g_1(r_1) - h$; by the observations above, $p \geqslant f(z) - h$. We now take two subcases.

Case 5, same assumption as Case 4, and

$$g_1(r_2) - h \geqslant \frac{r_2}{2r_1} p.$$

Then we also know that

$$g_2(r_2) - h \geqslant \frac{r_2}{2r_1}(f(z) - h);$$

this follows from plugging in the inequality for p and also the fact that $g_2 \geqslant g_1$. But recall that $g_2(r_2)$ corresponds to a feasible point in (8), and therefore $\rho^{(k)}(\zeta) \geqslant g_2(r_2)$. Thus, we have the bound

$$\rho^{(k)}(\zeta) - h \geqslant \frac{r_2}{2r_1}(f(z) - h).$$

Subtract $f(z) - h$ from both sides to obtain

$$\rho^{(k)}(\zeta) - f(z) \geqslant \frac{r_2 - 2r_1}{2r_1}(f(z) - h).$$

Plugging in the upper bound for r_1 and lower bound for r_2 yields

$$\rho^{(k)}(\zeta) - f(z) \geqslant \frac{4\min(\omega_k - b_1, b_2 - \lambda_k) - (9k)(b_2 - b_1)}{(9k)(b_2 - b_1)}(f(z) - h).$$

Using (11) yields

$$\rho^{(k)}(\zeta) - f(z) \geqslant \frac{29\min(\omega_k - b_1, b_2 - \lambda_k)}{(72k)(b_2 - b_1)}(f(z) - h).$$

This proves the theorem.

Case 6, same assumption as Case 4, and

$$g_1(r_2)-h\leqslant\frac{r_2}{2r_1}p.$$

In this case, we can apply Lemma 6 to conclude that the leading coefficient of g_1 is at least $p/(r_1\cdot(r_2-r_1))$ in magnitude. We can write an explicit formula for this coefficient; it is equal to

$$\frac{d_\sigma}{1-\phi}+\frac{d_\tau}{\phi}.$$

Thus we have the inequality

$$-\frac{d_\sigma}{1-\phi}-\frac{d_\tau}{\phi}\geqslant\frac{p}{r_1\cdot(r_2-r_1)}.$$

Multiply both sides by $\phi(1-\phi)$ and substitute the definition of r_1 to obtain

$$-\phi d_\sigma-(1-\phi)d_\tau\geqslant\frac{p}{(b_2-b_1)\cdot r_2}.$$

Next, observe that the leading coefficient of g_2 is equal to $d_\sigma+d_\tau$. Clearly we have

$$-d_\sigma-d_\tau\geqslant-\phi d_\sigma-(1-\phi)d_\tau$$

since d_σ, d_τ are nonpositive. Thus,

$$-d_\sigma-d_\tau\geqslant\frac{p}{(b_2-b_1)\cdot r_2}.$$

Now we apply Lemma 3 to conclude that the range of g_2 on interval $[0, r_2]$ is at least

$$\frac{pr_2}{4(b_2-b_1)}.$$

The range of g_2 is a lower bound on $\rho^{(k)}(\zeta)-f(z)$, so

$$\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)\geqslant\frac{pr_2}{4(b_2-b_1)}.$$

Now we use the fact that $p\geqslant f(z)-h$ and the lower bound on r_2 to obtain

$$\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)\geqslant\frac{\min(\omega_k-b_1,b_2-\lambda_k)}{18k(b_2-b_1)}(f(z)-h).$$

This concludes the proof of the theorem; all cases have been covered. □

4. The quadratic knapsack problem: Approximation

We now turn to the approximation algorithm Approx DP. The construction of the functions $\pi^{(k)}$ is fairly complicated; the various conditions in the theorem of the last section suggest that a simple-minded approximation scheme for $\mu^{(k)}$ might fail.

Recall that function $\pi^{(k)}$ is an approximation to $\mu^{(k)}$. The construction of $\pi^{(k)}$ is inductive: $\pi^{(1)}$ is set to $\mu^{(1)}$, and $\pi^{(k)}$ is constructed from $\pi^{(k-1)}$.

We assume that the variables are sorted so that

$$u_1 - l_1 \leqslant u_2 - l_2 \leqslant \cdots \leqslant u_n - l_n. \tag{24}$$

(Note that no such assumption was necessary for Exact DP.) Function $\pi^{(k)}$ will be piecewise quadratic and continuous, with each piece concave. Most pieces will be linear (which is a special case of concave). To construct $\pi^{(k)}$, we first construct an intermediate function $\hat{\pi}^{(k)}$ using Algorithm Scan-Breakpoint described in the last section. Scan-Breakpoint is applied to $\pi^{(k-1)}$ and q_k; thus,

$$\hat{\pi}^{(k)}(\zeta) = \min\{\pi^{(k-1)}(\zeta - t) + q_k(t) \colon t \in [l_k, u_k], \zeta - t \in [\lambda_{k-1}, \omega_{k-1}]\}. \tag{25}$$

Algorithm Scan-Breakpoint is applicable because, by the induction hypothesis, function $\pi^{(k-1)}$ is piecewise concave quadratic and continuous. Note that if we establish an upper bound on the number of breakpoints of $\pi^{(k-1)}$, then we can also bound the running time of Scan-Breakpoint. Running times will be discussed below.

From $\hat{\pi}^{(k)}$ we construct $\pi^{(k)}$ by making linear approximations. There are two cases: the first case is that (12) is satisfied when we take $j = k$.

In this case, we divide I_k with breakpoints. Specifically, we place a breakpoint at the center of I_k; call this b_0. Thus, $b_0 = \frac{1}{2}(\lambda_k + \omega_k)$. Note that for an interval $[b_1, b_2]$ to the left of b_0, $\omega_k - b_1$ will exceed $b_2 - \lambda_k$ in (11). The opposite holds to the right of b_0. Thus, the two halves are treated separately. First we place a series of breakpoints $b_1, b_2, \ldots, b_t$ between b_0 and ω_k geometrically spaced. Let $\delta > 0$ be a small parameter proportional to ε / n^2 (the exact formula for δ is given below). Define $p = 1 - \delta$, and let

$$b_i = b_0 + \tfrac{1}{2}(\omega_k - \lambda_k) \cdot (1 - p^i).$$

Notice that as i tends to infinity, p^i goes to zero and hence b_i tends to ω_k. Notice also that

$$b_i - b_{i-1} = \tfrac{1}{2}(\omega_k - \lambda_k) \cdot \delta \cdot p^{i-1}$$

whereas

$$\omega_k - b_{i-1} = \tfrac{1}{2}(\omega_k - \lambda_k) \cdot p^{i-1}.$$

Thus,

$$\frac{b_i - b_{i-1}}{\min(\omega_k - b_{i-1}, b_i - \lambda_k)} = \delta. \tag{26}$$

We continue the sequence until we reach b_t such that

$$b_t \geqslant \omega_k - u_1 + l_1. \tag{27}$$

How large does t have to be to satisfy this inequality? A calculation shows that if we pick

$$t \geqslant \frac{1}{|\ln p|} \ln\left[\frac{\omega_k - \lambda_k}{2(u_1 - l_1)}\right]$$

then we have b_t large enough. Using the facts that $p = 1 - \delta$ and $|\ln(1-\delta)| \geq \delta$ we can have the following upper bound on the right-hand side of the previous inequality:

$$\frac{1}{\delta} \cdot \ln\left[\frac{\omega_k - \lambda_k}{2(u_1 - l_1)}\right].$$

Once $b_1, \ldots, b_t$ are constructed, we change the value of b_t so that (27) holds as an equation. Note that this adjustment cannot increase the distance from b_{t-1} to b_t. Thus, for this interval, (26) holds if "$=$" is replace by "$\leq$". We will need only the inequality form of (26).

We construct a second sequence of breakpoints, $\hat{b}_1, \ldots, \hat{b}_t$ defined by

$$\hat{b}_i = b_0 - \tfrac{1}{2}(\omega_k - \lambda_k) \cdot (1 - p^i).$$

Similar inequalities as in the last paragraph hold for $\hat{b}_1, \ldots, \hat{b}_t$. Notice that with the same choice of t as above,

$$\hat{b}_t \leq \lambda_k + u_1 - l_1.$$

Again, we adjust $\hat{b}_t$ so that this inequality holds as an equation.

Once the breakpoints are selected, we define $\pi^{(k)}(\zeta)$ to be the piecewise linear function interpolated from $\hat{\pi}^{(k)}(\zeta)$ at the breakpoints $\{\hat{b}_t, \ldots, \hat{b}_1, b_0, b_1, \ldots, b_t\}$. In the interval $[\lambda_k, \hat{b}_t]$ and the interval $[b_t, \omega_k]$ we set $\pi^{(k)}$ exactly equal to $\hat{\pi}^{(k)}$ (no interpolation). See Figure 3 for an example of this construction. The algorithm to obtain $\pi^{(k)}$ from $\hat{\pi}^{(k)}$ by introducing breakpoints in this manner is called *Linearize*.

Note that the function $\pi^{(k)}$ is continuous and piecewise concave quadratic.

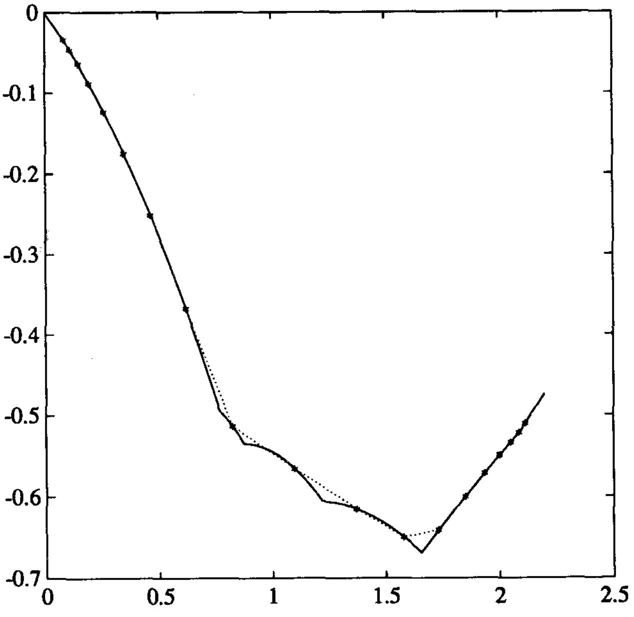

Fig. 3. Construction of $\pi^{(k)}$ (dotted) from $\hat{\pi}^{(k)}$ (solid).

The second case is that (12) is not satisfied for variable x_k. (Note that if (12) fails to be satisfied for a variable x_j; then we must have $j=k$ because of assumption (24).) In this case we construct $\pi^{(k)}$ in three pieces.

In the first piece, we artificially constrain x_k to lie between l_k and u'_k, where

$$u'_k = l_k + \sum_{i=1}^{k-1} (u_i - l_i).$$

Note that (12) is satisfied with this bound on x_k. Notice also that if ζ lies between λ_k and m, where

$$m = u_1 + \cdots + u_{k-1} + l_k$$

then no value of x_k larger than u'_k is feasible in (8). Thus, for ζ in the interval $[\lambda_k, m]$, this artificial upper bound on x_k is not actually a restriction (in particular, $\mu^{(k)}$, $\rho^{(k)}$, and $\hat{\pi}^{(k)}$ are unchanged).

This interval $[\lambda_k, m]$ is the first piece. We use Linearize to construct a piecewise-linear $\pi^{(k)}$ from $\hat{\pi}^{(k)}$ in the interval $[\lambda_k, \omega'_k]$, where

$$\omega'_k = u_1 + \cdots + u_{k-1} + u'_k.$$

A calculation shows that b_0 chosen by the Linearize algorithm turns out to be exactly m in the previous paragraph. Once Linearize is finished, we discard the breakpoints to the right of b_0, and define $\pi^{(k)}$ on $[\lambda_k, m]$ as the resulting piecewise-linear function.

The second piece is the interval $[m, m']$ where

$$m' = l_1 + \cdots + l_{k-1} + u_k.$$

(Note that $m' \geqslant m$: this follows from the assumption that (12) fails.) In this interval, we define $\pi^{(k)}$ to be equal to $\hat{\pi}^{(k)}$.

The third piece is the interval $[m', \omega_k]$. In this piece we artificially constrain x_k to lie between l'_k and u_k, where

$$l'_k = u_k - \sum_{i=1}^{k-1} (u_i - l_i).$$

Once again, with this definition (12) is satisfied. Notice moreover that if $\zeta \in [m', \omega_k]$ then no value of x_k smaller than l'_k could actually be feasible for (8). Thus, as before, the restriction of x_k to $[l'_k, u_k]$ does not affect $\mu^{(k)}$, $\rho^{(k)}$ or $\hat{\pi}^{(k)}$ for $\zeta \in [m', \omega_k]$. We apply algorithm Linearize to come up with $\pi^{(k)}$ on the interval $[\lambda'_k, \omega_k]$, where

$$\lambda'_k = l_1 + \cdots + l_{k-1} + l'_k.$$

Notice that b_0 constructed by Linearize is coincident with m'. On the interval $[m', \omega_k]$ we define $\pi^{(k)}$ to be the linear function constructed by Linearize on the restricted problem.

The algorithm for constructing $\pi^{(k)}$ from $\hat{\pi}^{(k)}$ in the case that (12) fails is called *Linearize-3*. Thus, Linearize-3 is the algorithm described in the last few paragraphs that works in three pieces. An example of Linearize-3 is illustrated in Figure 4.

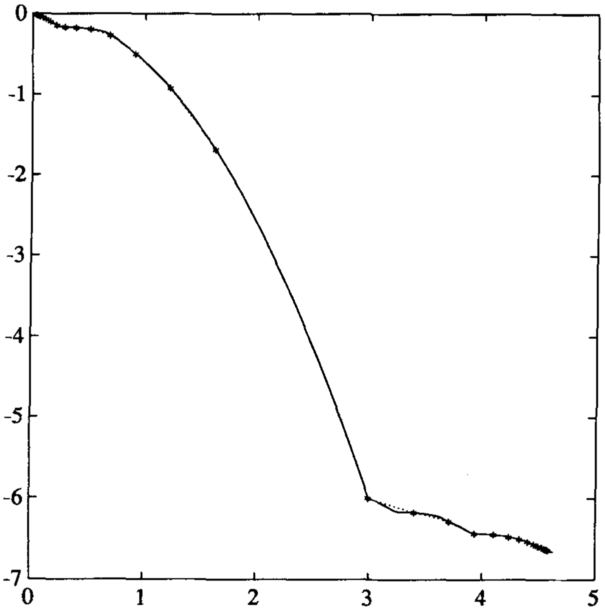

Fig. 4. Construction of $\pi^{(k)}$ (dotted) from $\hat{\pi}^{(k)}$ (solid) when (12) fails.

The goal of this construction is to show that $\pi^{(k)}$ approximates $\mu^{(k)}$ in some sense. Before proving the main theorem, we first consider what happens to ratios when their numerators and denominators are interpolated. The proof of the following lemma is straightforward and appears in Vavasis (1991).

Lemma 8. *Let n_1, n_2, d_1, d_2 be real numbers such that d_1, d_2 are positive. Suppose $\phi \in [0, 1]$. Then*

$$\min\left(\frac{n_1}{d_1}, \frac{n_2}{d_2}\right) \leqslant \frac{(1-\phi)n_1+\phi n_2}{(1-\phi)d_1+\phi d_2} \leqslant \max\left(\frac{n_1}{d_1}, \frac{n_2}{d_2}\right). \qquad \square$$

Theorem 9. *We have the following bound for $\pi^{(k)}$, $k = 1, \ldots, n$, and $\zeta \in I_k$:*

$$|\pi^{(k)}(\zeta) - \mu^{(k)}(\zeta)| \leqslant s_k \cdot (\rho^{(k)}(\zeta) - \mu^{(k)}(\zeta)) \tag{28}$$

where

$$s_k = 144kn\delta,$$

provided k, n, δ are such that $144kn\delta \leqslant 1$.

Proof. This will be proved by induction on k. The base case $k = 1$ is trivial, since $\pi^{(1)}$, $\rho^{(1)}$, and $\mu^{(1)}$ are identical functions. We now address the case $k \geqslant 2$, where (28) is assumed to hold in the $k-1$ case. First, we examine the difference between

$\hat{\pi}^{(k)}$ and $\mu^{(k)}$. Fix a particular $\zeta \in I_k$. Referring to (9), suppose the minimum for ζ is achieved at a particular t_1, i.e.,

$$\mu^{(k-1)}(\zeta - t_1) + q_k(t_1) = \mu^{(k)}(\zeta).$$

Since $\hat{\pi}^{(k)}$ is defined by (25), there is a t_2 such that

$$\pi^{(k-1)}(\zeta - t_2) + q_k(t_2) = \hat{\pi}^{(k)}(\zeta).$$

Now we do a calculation, taking into account the inductive hypothesis and also the fact that t_1 is chosen to be a minimizer:

$$\begin{aligned}\mu^{(k)}(\zeta) &= \mu^{(k-1)}(\zeta - t_1) + q_k(t_1)\\ &\leq \mu^{(k-1)}(\zeta - t_2) + q_k(t_2)\\ &\leq \pi^{(k-1)}(\zeta - t_2) + s_{k-1}[\rho^{(k-1)}(\zeta - t_2) - \mu^{(k-1)}(\zeta - t_2)] + q_k(t_2)\\ &\leq \hat{\pi}^{(k)}(\zeta) + s_{k-1}[\rho^{(k-1)}(\zeta - t_2) - \mu^{(k-1)}(\zeta - t_2)].\end{aligned}$$

Next, we bound the bracketed expression above:

$$\begin{aligned}\rho^{(k-1)}(\zeta - t_2) - \mu^{(k-1)}(\zeta - t_2) &= (\rho^{(k-1)}(\zeta - t_2) + q_k(t_2))\\ &\qquad - (\mu^{(k-1)}(\zeta - t_2) + q_k(t_2))\\ &\leq \rho^{(k)}(\zeta) - \mu^{(k)}(\zeta).\end{aligned}$$

Combining the two previous chains of inequalities shows that

$$\mu^{(k)}(\zeta) - \hat{\pi}^{(k)}(\zeta) \leq s_{k-1} \cdot (\rho^{(k)}(\zeta) - \mu^{(k)}(\zeta)).$$

A similar bound is proved symmetrically for $\hat{\pi}^{(k)}(\zeta) - \mu^{(k)}(\zeta)$. Thus, we conclude that

$$|\mu^{(k)}(\zeta) - \hat{\pi}^{(k)}(\zeta)| \leq s_{k-1} \cdot (\rho^{(k)}(\zeta) - \mu^{(k)}(\zeta)). \tag{29}$$

Thus, we have a bound on the difference between $\mu^{(k)}$ and $\hat{\pi}^{(k)}$. If ζ does not lie in the linear pieces constructed by Linearize or Linearize-3, then the two values $\pi^{(k)}(\zeta)$ and $\hat{\pi}^{(k)}(\zeta)$ are identical. Thus, if $\hat{\pi}^{(k)}(\zeta) = \pi^{(k)}(\zeta)$ then (29) proves the theorem, since $s_{k-1} \leq s_k$.

The more complicated situation is that ζ happens to lie between two consecutive breakpoints, say b_1 and b_2. Choose $\phi \in [0, 1]$ so that

$$\zeta = (1-\phi)b_1 + \phi b_2.$$

Let h denote the interpolated value of $\mu^{(k)}$, that is, $h = (1-\phi)\mu^{(k)}(b_1) + \phi\mu^{(k)}(b_2)$. Note that by definition of $\pi^{(k)}$ we have the relation

$$\pi^{(k)}(\zeta) = (1-\phi)\hat{\pi}^{(k)}(b_1) + \phi\hat{\pi}^{(k)}(b_2).$$

We first take the case when (12) holds and Linearize was used. By construction of the breakpoints, we know that

$$\frac{b_2-b_1}{\min(\omega_k-b_1,\, b_2-\lambda_k)}\leq\delta.$$

Therefore by Theorem 7 we conclude that

$$|\mu^{(k)}(\zeta)-h|\leq 72k\cdot\delta\cdot(\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)). \tag{30}$$

Notice that in (29) we showed that

$$\frac{\mu^{(k)}(b_i)-\hat{\pi}^{(k)}(b_i)}{\rho^{(k)}(b_i)-\mu^{(k)}(b_i)}\leq s_{k-1}$$

for $i=1,2$. The same bound holds if the two terms in the numerator are interchanged, also for $i=1,2$. We can interpolate the numerator and denominator of the left-hand side of the preceding inequality between b_1 and b_2, applying Lemma 8, to conclude that

$$\frac{|h-\pi^{(k)}(\zeta)|}{\rho_1-h}\leq s_{k-1}$$

where $\rho_1=(1-\phi)\rho^{(k)}(b_1)+\phi\rho^{(k)}(b_2)$. Rewriting this, we have

$$|h-\pi^{(k)}(\zeta)|\leq s_{k-1}\cdot(\rho_1-h).$$

Now we add $s_{k-1}(h-\mu^{(k)}(\zeta))$ on the right-hand side, using (30). We also note that since $\rho^{(k)}$ is globally concave, $\rho_1\leq\rho^{(k)}(\zeta)$. Thus, we obtain

$$\begin{aligned}|h-\pi^{(k)}(\zeta)| &\leq s_{k-1}(\rho^{(k)}(\zeta)-h)\\ &\leq s_{k-1}(\rho^{(k)}(\zeta)-h)+s_{k-1}(h-\mu^{(k)}(\zeta))\\ &\quad+s_{k-1}\cdot(72k)\delta(\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta))\\ &\leq s_{k-1}(1+72k\delta)(\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)).\end{aligned} \tag{31}$$

Since the assumption stated in the theorem is that $s_{k-1}\leq 1$, we have

$$s_{k-1}(1+72k\delta)\leq s_{k-1}+72k\delta.$$

Now we add (30) and (31), applying the triangle inequality on the left, to obtain

$$|\mu^{(k)}(\zeta)-\pi^{(k)}(\zeta)|\leq(s_{k-1}+144k\delta)\cdot(\rho^{(k)}(\zeta)-\mu^{(k)}(\zeta)),$$

which proves the theorem, by definition of s_k.

The last case is ζ lies between two breakpoints, b_1, b_2, linearized by Linearize-3, say for example, in the first piece computed by Linearize-3 (the third piece is handled analogously, and the second piece needs no further analysis since $\pi^{(k)}=\hat{\pi}^{(k)}$ on the second piece). In this case, all the calculations of the previous case hold, except the derivation of (30) is more complicated. Specifically, we can obtain (30) by applying Theorem 7 to the problem in which x_k is restricted to $[l_k, u'_k]$ where u'_k was defined above in the description of Linearize-3. For the restricted problem, (12) holds. Moreover, the breakpoint distances are defined in terms of the restricted problem, so we get (30) for the restricted problem. Finally, we observe (as noted above) that $\mu^{(k)}$ and $\rho^{(k)}$ are unchanged when x_k is restricted. □

Recall that the goal of this whole procedure was to show that $\pi^{(n)}(\gamma)$ is within $\varepsilon \cdot (\rho^{(n)}(\gamma) - \mu^{(n)}(\gamma))$ of $\mu^{(n)}(\gamma)$. This bound will hold, according to the preceding theorem, provided that δ is chosen so that

$$\varepsilon \geqslant 144n^2\delta,$$

i.e., $\delta \leqslant \varepsilon/(144n^2)$.

This completes the description of algorithm Approx DP for the concave case of (6). In the next section we discuss the extension to the indefinite case. We cover two additional topics in this section for the concave case: identification of the vector $x^{\diamond}$ and running time.

First, we discuss the construction of $x^{\diamond}$. Note that algorithm Approx DP returns only an approximate value of the optimum solution, not an approximation vector. There is a standard technique in dynamic programming to obtain the optimizing vector from the optimal solution. Specifically, the DP algorithm is augmented so that it tags the data items computed at each step with their source from the previous step. In the particular case of Approx DP, the tags take the form of a vector $x^{(k)}(b_i)$, such that the optimal objective function of (8) is approximately attained at this vector when $\zeta = b_i$. Here, b_i is a breakpoint of $\pi^{(k)}$. Thus, by saving one vector per breakpoint, an approximately optimal solution may be constructed.

Some technical difficulties are encountered in the analysis of how far $f(x^{(k)}(b_i))$ lies from $\pi^{(k)}(b_i)$. These difficulties stem from the fact that no bound has been derived on $|\pi^{(k)}(\zeta) - \hat{\pi}^{(k)}(\zeta)|$ in the preceding analysis. This requires the construction of $x^{(k)}(b_i)$ to be more complicated than might be expected. The details are in Vavasis (1991).

The last topic before turning to the indefinite case is the running time. The main work of Approx DP is n calls to Scan-Breakpoint. A straightforward implementation of Scan-Breakpoint would require time proportional to $r^2 \log r$, where r is the number of breakpoints of $\pi^{(k-1)}$. In this implementation, the various concave quadratic functions on $[\lambda_k, \omega_k]$ would be written out explicitly — their number would be proportional to the number of breakpoints of $\pi^{(k-1)}$. Then the pairwise intersection between every pair would be computed (at most $O(r^2)$ intersections) to determine new breakpoints. The new list of breakpoints would be sorted (requiring $r^2 \log r$ steps), and then $\pi^{(k)}$ would be written down.

It turns out that Scan-Breakpoint can be implemented much more efficiently using an algorithm by Hershberger (1989), which finds the lower envelope of parabolic segments. The running time obtained by Hershberger is $O(n \cdot \alpha(n) \cdot \log n)$, where $\alpha(n)$ denotes the "inverse Ackerman function". This function grows extremely slowly; for any problem size that might ever be computed, it is safe to assume $\alpha(n) \leqslant 6$. Hershberger also gives more detail about a straightforward divide-and-conquer approach, including running time estimates.

Thus, to bound the running time it suffices to bound the number of breakpoints in $\pi^{(k)}$. There are three kinds of breakpoints in $\pi^{(k)}$; we will bound each kind separately.

The Type I breakpoints are those introduced by Linearize or Linearize-3. An upper bound on the number of these breakpoints was derived above; the bound is

$$\frac{2}{\delta} \ln\left(\frac{\omega_n - \lambda_n}{2(u_1 - l_1)}\right).$$

The Type II breakpoints are those occurring in the intervals $[\lambda_k, \hat{b}_t]$ and $[b_t, \omega_k]$. We claim that there are at most $O(k)$ of these breakpoints. We skip the details of this argument (see Vavasis, 1991).

Type III breakpoints are the breakpoints of $\pi^{(k)}$ occurring between m and m' in Linearize-3. It can be shown using an argument analogous to Theorem 4 that $\pi^{(k)}(\zeta)$ for $\zeta \in (m, m')$ will have the form

$$\pi^{(k-1)}(b) + q_k(\zeta - b), \tag{32}$$

where b is a breakpoint of $\pi^{(k-1)}$. It cannot have the form $\pi^{(k-1)}(\zeta - b) + q_k(b)$, where $b = l_k$ or $b = u_k$, because $\zeta - b \notin [\lambda_{k-1}, \omega_{k-1}]$ in this case. For example, we can compute that

$$\zeta - l_k > m - l_k \geqslant u_1 + \cdots + u_{k-1}$$

by definition of m. Thus, $\pi^{(k)}(\zeta)$ is one of l possible concave quadratic functions, where l is the number of breakpoints of $\pi^{(k-1)}$. A more careful analysis shows that each quadratic function can occur in only one piece, so that the total number of Type III breakpoints is bounded by l, and, moreover, that b in (32) cannot be a Type II or Type III breakpoint of $\pi^{(k-1)}$ (only a Type I breakpoint). The details are in Vavasis (1991).

Thus, the total number of breakpoints is asymptotically dominated by r, the number of Type I breakpoints. The running time of Scan-Breakpoints is proportional to $r\alpha(r)\log r$, and the total running time is dominated by n calls to Scan-Breakpoints. This gives a total running time estimate of

$$O(nr\alpha(r) \log r)$$

where

$$r = \frac{288n^2}{\varepsilon} \cdot \ln\left(\frac{\omega_n - \lambda_n}{2(u_1 - l_1)}\right).$$

If we also want to compute the vector $x^{(n)}(\gamma)$, another term of the form $n^2 r$ is added to the running time.

We remark that this running time is bounded by a polynomial in n/ε and in the size of the problem. In particular, the logarithmic factor is bounded by the number of bits to write the problem assuming that all the input data for the problem is integral. This is known as a "weakly polynomial" algorithm, meaning that the number of arithmetic operations depends on the number of bits in the problem.

5. The quadratic knapsack problem: Indefinite problems

Our next task is to extend Approx DP to handle general indefinite problems. Given an indefinite instance of (6), we split the variables into two vectors $\boldsymbol{y}$ and $\boldsymbol{z}$, convex and concave, as follows:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{y}^{\mathrm{T}}C\boldsymbol{y}+\boldsymbol{c}^{\mathrm{T}}\boldsymbol{y}+\boldsymbol{z}^{\mathrm{T}}D\boldsymbol{z}+\boldsymbol{d}^{\mathrm{T}}\boldsymbol{z} \\ \text{subject to} & y_1+\cdots+y_{n_1}+z_1+\cdots+z_{n_2}=\gamma, \\ & l_i\leqslant y_i\leqslant u_i, \quad i=1,\ldots,n_1, \\ & l'_i\leqslant z_i\leqslant u'_i, \quad i=1,\ldots,n_2. \end{array} \tag{33}$$

Here, C is a positive semidefinite matrix and D is a negative semidefinite matrix. Note that the linear variables (those whose quadratic coefficients are zero) may be assigned to either $\boldsymbol{y}$ or $\boldsymbol{z}$.

The next step is to compute an approximation for the concave part. This is done using the algorithm in the last section. In particular, we obtain a function $\pi(\zeta)$ defined on the interval

$$[l'_1+\cdots+l'_{n_2}, u'_1+\cdots+u'_{n2}]$$

such that

$$|\pi(\zeta)-\mu(\zeta)|\leqslant\varepsilon\cdot(\rho(\zeta)-\mu(\zeta))$$

where

$$\begin{array}{lll} \mu(\zeta) & = & \text{minimum of} \quad \boldsymbol{z}^{\mathrm{T}}D\boldsymbol{z}+\boldsymbol{d}^{\mathrm{T}}\boldsymbol{z} \\ & & \text{subject to} \quad z_1+\cdots+z_{n_2}=\zeta, \\ & & \qquad\qquad l'_i\leqslant z_i\leqslant u'_i, \quad i=1,\ldots,n_2, \end{array}$$

and

$$\begin{array}{lll} \rho(\zeta) & = & \text{maximum of} \quad \boldsymbol{z}^{\mathrm{T}}D\boldsymbol{z}+\boldsymbol{d}^{\mathrm{T}}\boldsymbol{z} \\ & & \text{subject to} \quad z_1+\cdots+z_{n_2}=\zeta, \\ & & \qquad\qquad l'_i\leqslant z_i\leqslant u'_i, \quad i=1,\ldots,n_2. \end{array}$$

Once we have π, then we also compute a function ψ for the convex part defined by

$$\begin{array}{lll} \psi(\theta) & = & \text{minimum of} \quad \boldsymbol{y}^{\mathrm{T}}C\boldsymbol{y}+\boldsymbol{c}^{\mathrm{T}}\boldsymbol{y} \\ & & \text{subject to} \quad y_1+\cdots+y_{n_1}=\theta, \\ & & \qquad\qquad l_i\leqslant y_i\leqslant u_i, \quad i=1,\ldots,n_1. \end{array}$$

We remark that ψ can be exactly computed in $\mathrm{O}(n_1 \log n_1)$ time. Function ψ has the following properties: it is piecewise quadratic and (globally) convex, with $\mathrm{O}(n_1)$ breakpoints. The algorithm to compute ψ is described in detail in Vavasis (1992b). Briefly, the idea is as follows. For a fixed θ, this problem is a convex minimization problem and hence is solved when the KKT conditions are satisfied. There is only on KKT multiplier of importance, namely, the multiplier λ for the equation constraint. It is possible to deduce values for all of the variables, and hence also for the objective function, given λ. Moreover, it is possible to explicitly solve this problem (first observed by Helgason, Kennington and Lall, 1980) by solving for λ. If θ is indeterminate, then one can write down an explicit monotonic dependence of λ upon θ, and hence of the optimal objective function upon θ — this dependence is the function ψ.

From π and ψ we now can approximately solve the indefinite problem (33). Specifically, we have to minimize the sum

$$\pi(\zeta)+\psi(\gamma-\zeta)$$

for all choices of ζ feasible for π such that $\gamma-\zeta$ is feasible for ψ. This is done by looping over the breakpoints of π.

When the minimizer $\zeta^{\diamond}$ is identified in the preceding formula, we can claim that we have an ε-approximate minimum for the whole problem. Write the approximate minimum as $q^{\diamond}$:

$$q^{\diamond}=\pi(\zeta^{\diamond})+\psi(\gamma-\zeta^{\diamond}).$$

This is an overall ε-approximate minimum. See Vavasis (1991) for the details of this claim. Thus, we have proved the following theorem:

Theorem 10. *Consider an indefinite quadratic knapsack problem of the form* (33). *Then an ε-approximate minimum can be computed in time proportional to*

$$\mathrm{O}(n_1 \log n_1 + n_2 r\alpha(r) \log r)$$

where

$$r=\frac{288n_2^2}{\varepsilon}\ln\left(\frac{\omega_{n_2}-\lambda_{n_2}}{2(u_1'-l_1')}\right). \qquad \square$$

6. Dependence of the running time on the parameters

In this section we discuss our two main theorems (Theorem 2 and Theorem 10). We give some indication why the dependence on ε and t or n that we obtained in the main theorems might be expected.

First, we address the dependence of Theorem 2 on ε. To simplify this discussion, let us restrict attention to the case $t = 1$, for example, a quadratic objective function of the form $cy^2 + f^T z$, where y is a scalar unknown and $c < 0$. Suppose that we were able to obtain an approximation algorithm with running time dependence on ε better than $1/\varepsilon$. Suppose, for example, that we had an approximation algorithm whose running time were polynomial in $|\log \varepsilon|$. It is known that if a point is sufficiently close to optimal for an instance of quadratic programming — in particular, within $2^{-O(L)}$ of optimal — then an exact optimum may be found in polynomial time. Here, L denotes the number of bits needed to write the problem. If there were an approximation algorithm whose running time were polynomial in $|\log \varepsilon|$, then we set $\varepsilon = 2^{-O(L)}$ and in polynomial time come up with an exact solution to the concave problem with objective function $cy^2 + f^T z$.

This would seem to contradict recent results by Pardalos and Vavasis (1991), who proved that problems of this form are NP-hard. Thus, assuming $P \neq NP$, polynomial dependence on $|\log \varepsilon|$ is not expected, and therefore polynomial dependence on $1/\varepsilon$ seems like it might be the best possible.

The same remarks apply to Theorem 10. If it were possible to obtain polynomial time dependence on $|\log \varepsilon|$ and polynomial in the rest of the problem size, then concave quadratic knapsack problems could be solved in polynomial time, apparently contradicting Sahni's result that these problems are NP-hard.

Next, we investigate the exponential dependence on t in Theorem 2. We would prefer to have polynomial dependence on n (and therefore t also), as in Theorem 10. Suppose there were an approximation algorithm for the general indefinite problem whose running time were polynomial in t and $1/\varepsilon$. The existence of such an algorithm would imply that $P = NP$; the proof of this was given by Vavasis (1992a). This suggests that polynomial dependence on t is not possible for the general case.

7. Open questions

Probably the most interesting question is whether the results on the knapsack problem extend to more general optimization problems. One promising area is nonconvex discrete-time optimal control, in which the problem can be decomposed into steps in the same way we decomposed the knapsack problem.

Another open question is whether the running-time bound in Section 5 can be improved to a strongly polynomial bound.

Acknowledgement

The author wishes to thank the anonymous referee who suggested the use of simultaneous diagonalization to strengthen the original version of Theorem 2.

References

R.E. Bellman, *Dynamic Programming* (Princeton University Press, Princeton, NJ, 1957).

P. Brucker, "An O(n) algorithm for quadratic knapsack problems," *Operations Research Letters* 3 (1984) 163-166.

R.W. Cottle, S.G. Duvall and K. Zikan, "A Lagrangean relaxation algorithm for the constrained matrix problem," *Naval Research Logistics Quarterly* 33 (1986) 55-76.

R.M. Freund, R. Roundy, and M.J. Todd, "Identifying the set of always active constraints in a system of linear inequalities by a single linear program," Working Paper 1674-85, Sloan School of Management, MIT (Cambridge, MA, 1985).

P.E. Gill, W. Murray and M.H. Wright, *Practical Optimization* (Academic Press, London, 1981).

G.H. Golub and C.F. Van Loan, *Matrix Computations* (Johns Hopkins University Press, Baltimore, MD, 1989, 2nd ed.).

R. Helgason, J. Kennington and H. Lall, "A polynomially bounded algorithm for a singly constrained quadratic program," *Mathematical Programming* 18 (1980) 338-343.

J. Hershberger, "Finding the upper envelope of n line segments in O(n log n) time," *Information Processing Letters* 33 (1989) 169-174.

S. Kapoor and P.M. Vaidya, "Fast algorithms for convex quadratic programming and multicommodity flows," in: *Proceedings of the 18th Annual ACM Symposium on Theory of Computing* (ACM Press, New York, 1986) pp. 147-159.

M.K. Kozlov, S.P. Tarasov and L.G. Hačijan, "Polynomial solvability of convex quadratic programming," *Doklad Akademii Nauk SSSR* 248 (1979) 1049-1051. [Translated in: *Soviet Mathematics Doklady* 20 (1979) 1108-1111.]

L. Lovász, *An Algorithmic Theory of Numbers, Graphs and Convexity* (SIAM, Philadelphia, PA, 1986).

J.J. Moré and S.A. Vavasis, "On the solution of concave knapsack problems," *Mathematical Programming* 49 (1991) 397-411.

K.G. Murty and S.N. Kabadi, "Some NP-complete problems in quadratic and nonlinear programming," *Mathematical Programming* 39 (1987) 117-129.

A.S. Nemirovsky and D.B. Yudin, *Problem Complexity and Method Efficiency in Optimization* (Wiley, Chichester, 1983). [Translated by E.R. Dawson from *Slozhnost' Zadach i Effektivnost' Metodov Optimizatsii* (1979).]

C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization:Algorithms and Complexity* (Prentice-Hall, Englewood Cliffs, NJ, 1982).

P.M. Pardalos and J.B. Rosen, *Constrained Global Optimization: Algorithms and Applications*, Lecture Notes in Computer Science No. 268 (Springer, Berlin, 1987).

P.M. Pardalos and S.A. Vavasis, "Quadratic programming with one negative eigenvalue is NP-hard," *Journal of Global Optimization* 1 (1990) 15-22.

S. Sahni, "Computationally related problems," *SIAM Journal on Computing* 3 (1974) 262-279.

P.M. Vaidya, "Speeding-up linear programming using fast matrix multiplication (extended abstract)," *Proceedings of the 30th Symposium on Foundations of Computer Science* (ACM Press, New York, 1989) pp. 332-337.

S.A. Vavasis, "Quadratic programming is in NP," *Information Processing Letters* 36 (1990) 73-77.

S.A. Vavasis, "Approximation algorithms for indefinite quadratic programming," Technical Report 91-1228, Department of Computer Science, Cornell University (Ithaca, NY, 1991).

S.A. Vavasis, "On approximation algorithms for concave quadratic programming," in: C.A. Floudas and P.M. Pardalos, eds., *Recent Advances in Global Optimization* (Princeton University Press, Princeton, NJ, 1992a) pp. 3-18.

S.A. Vavasis, "Local minima for indefinite quadratic knapsack problems," *Mathematical Programming* 54 (1992b) 127-153.

Y. Ye and E. Tse, "An extension of Karmarkar's projective algorithm for convex quadratic programming," *Mathematical Programming* 44 (1989) 157-179.