SIAM J. DISCRETE MATH.
Vol. 5, No. 1, pp. 127–143, February 1992

011

ON TENSOR POWERS OF INTEGER PROGRAMS*

ROBIN PEMANTLE†, JAMES PROPP‡, AND DANIEL ULLMAN§

Abstract. A natural product on integer programming problems with nonnegative coefficients is defined. Hypergraph covering problems are a special case of such integer programs, and the product defined is a generalization of the usual hypergraph product. The main theorem of this paper gives a sufficient condition under which the solution to the nth power of an integer program is asymptotically as good as the solution to the same nth power when the variables are not necessarily integral but may be arbitrary nonnegative real numbers.

Key words. integer program, linear program, hypergraph, covering

AMS(MOS) subject classifications. 90C10; secondary, 05C65

1. Definitions and notations. The minimization problems that we consider here are of the form "Minimize the quantity

$$c_1x_1 + c_2x_2 + \cdots + c_dx_d$$

subject to the constraints

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1d}x_d &\geq b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2d}x_d &\geq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{md}x_d &\geq b_m, \end{aligned}$$

where a_{ij}, b_i, and c_j are fixed nonnegative real numbers, and x_j are unknown nonnegative integers." Label the constraints $C(1), \cdots, C(m)$. We would lose no generality by throwing out those variables x_j for which $c_j = 0$ (together with every constraint $C(i)$ for which $a_{ij} > 0$) and those constraints for which $b_i = 0$, thus making all b_i and c_j positive. For the time being, however, we do not require positivity.

We may write our integer program more compactly as "Minimize c^Tx subject to $Ax \geq b$ with $x \geq 0$ and integral," where A is a nonnegative m-by-d matrix, b is a nonnegative column vector of length m, c is a nonnegative column vector of length d, and x ranges over the set of nonnegative integer column vectors of length d. Assume further that $b_i > 0$ implies the existence of a j for which $a_{ij} > 0$. We denote this integer program by the triple $P = (A, b, c)$. Our positivity assumptions on A, b, and c imply that feasible solution vectors x exist; the minimum possible value of c^Tx as x ranges over all solution vectors is called the *value* of P, denoted $v(P)$.

Associated with the integer program P is its *LP-relaxation*, obtained by dropping the requirement that the entries in the solution vector be integers. We let $v^*(P)$ (the

* Received by the editors May 22, 1989; accepted for publication (in revised form) January 10, 1991.

† University of California at Berkeley, Berkeley, California 94720. Present address, Oregon State University, Corvallis, Oregon 97331. This author was supported by a National Science Foundation post-doctoral fellowship.

‡ University of California at Berkeley, Berkeley, California 94720. Present address, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

§ The George Washington University, Washington, DC 20052. This author was supported by a Junior Scholar Incentive Award from The George Washington University.

LP-relaxed value of P) signify the optimum of this relaxed linear program. Note that $v^*(P)$ is a real number between 0 and $v(P)$.

Also associated with the minimization program P is the program $P^\perp$, "Maximize $b^T y$ subject to $A^T y \leq c$, with $y \geq 0$ and integral." The program $P^\perp$ is called the *dual* of P. The well-known duality theorem asserts that the optimum values of the respective LP-relaxations of P and $P^\perp$ are equal; that is, if we extend our definitions of v and v^* to cover maximization programs in the natural way, we have $v^*(P^\perp) = v^*(P)$. However, it is by no means true that $v(P^\perp) = v(P)$; for, in general, we have

$$0 \leq v(P^\perp) \leq v^*(P^\perp) = v^*(P) \leq v(P),$$

so that if b and c have integer entries, but $v^*(P)$ is not an integer, there is no chance of the integers $v(P^\perp)$ and $v(P)$ being equal.

Given two minimization programs P and P', there is natural way to define two other programs called their *sum* and *tensor product*. (For wholly analogous constructions in information theory, see pp. 65–66 of [7]. See also [1], where the analogous sum of two network flow problems is seen to correspond to parallel-composition of the two networks.) Suppose $P = (A, b, c)$, $P' = (A', b', c')$, where A is m-by-d and A' is m'-by-d'. We define $A \oplus A'$ as the $(m + m')$-by-$(d + d')$ matrix

$$\begin{pmatrix} A & 0 \\ 0 & A' \end{pmatrix},$$

$b \oplus b'$ as the vector of length $m + m'$ obtained by concatenating the vectors b and b', and $c \oplus c'$ as a similar concatenation; we then define the sum $P \oplus P'$ of the programs P and P' to be the program $(A \oplus A', b \oplus b', c \oplus c')$. To define multiplication of programs, it is notationally convenient to allow indices for vectors and matrices to be not just natural numbers, but also pairs of natural numbers; then, the tensor product of A and A' may be defined as the matrix whose $((i, j), (k, l))$th entry is $a_{ik} a'_{jl}$. (If, as is often done, we re-index the product so that the indices are natural numbers, then the matrix $A \otimes A'$ may be depicted as

$$\begin{pmatrix} a_{1\,1}A' & a_{1\,2}A' & \cdots & a_{1\,n}A' \\ a_{2\,1}A' & a_{2\,2}A' & \cdots & a_{2\,n}A' \\ \vdots & \vdots & & \vdots \\ a_{m\,1}A' & a_{m\,2}A' & \cdots & a_{m\,n}A' \end{pmatrix};$$

however, this representation is not necessary for our purposes.) We define $b \otimes b'$ as the column vector of length mm' whose (i, j)th entry is $b_i b'_j$, and $c \otimes c'$ as the column vector of length nn' whose (k, l)th entry is $c_k c'_l$. We conclude by defining the product $P \otimes P'$ of the programs P and P' to be the program $(A \otimes A', b \otimes b', c \otimes c')$.

We leave it to the reader to verify that $\oplus$ and $\otimes$ satisfy the natural commutativity, associativity, and distributivity properties; moreover, we can define the "empty program" (no variables, no constraints) and the "identity program" (with A as the 1-by-1 identity matrix, and b and c as vectors whose lone entry is 1) to serve as identity elements for $\oplus$ and $\otimes$, respectively. We further remark that, defining $\oplus$ and $\otimes$ for maximization programs in the obvious way, we have $(P \oplus P')^\perp = P^\perp \oplus P'^\perp$ and $(P \otimes P')^\perp = P^\perp \otimes P'^\perp$. Finally, we point out that if P is a minimization program in which some of the entries of the b-vector or c-vector equal 0, there is a canonical program P' obtained by throwing out the corresponding variables and constraints;

moreover, the mapping $P \mapsto P'$ preserves $v(\cdot)$ and $v^*(\cdot)$ and commutes with the operations $\oplus$ and $\otimes$. Hence, in the following we may without loss of generality assume that b_i and c_j are positive for all i, j and that $v(P) > 0$.

An easy fact from the next section is that $v^*(P \otimes P') = v^*(P)v^*(P')$; however, an example there will show that it is not true in general that $v(P \otimes P') = v(P)v(P')$, and that we must content ourselves with the weaker statement $v(P \otimes P') \leq v(P)v(P')$. If we define $P^{\otimes n}$ as $P \otimes P \otimes \cdots \otimes P$ with n occurrences of P, then this inequality implies that $v(P^{\otimes i+j}) \leq v(P^{\otimes i})v(P^{\otimes j})$ for all i, j; by Fekete's lemma [2], we conclude that as n gets large, the quantity

$$\sqrt[n]{v(P^{\otimes n})}$$

approaches its infimum, which we call the *asymptotic optimum value* of P. The following theorem gives conditions on P that force the asymptotic optimum value to equal the value of the LP-relaxation of P.

THEOREM 1. *Let $P = (A, b, c)$ be an integer minimization program in which b_i and c_j are strictly positive for all i and j. Suppose there exists an optimum solution-vector $(\alpha(1), \alpha(2), \cdots, \alpha(d))$ for the* LP*-relaxation of P such that*

$$\prod_j \left(\frac{a_{ij}}{b_i}\right)^{a_{ij}\alpha(j)/v^*(P)} \leq 1 \text{ for all } i. \tag{1}$$

Then $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ as $n \to \infty$.

(In condition (1), we are to take $0^0 = 1$, as usual. If we multiply the exponent $V^*(P)$, the resulting inequality is equivalent to (1) and looks simpler, but the form we have given will be more useful.)

It has already been mentioned (see the first paragraph of §1) that once we have assumed that our program P satisfies $a_{ij}, b_i, c_j \geq 0$ for all i, j, we may as well assume that $b_i > 0$ for all i and $c_j > 0$ for all j. An explanation of the role played by condition (1) will be given later. For now, let us mention the following result.

THEOREM 2. *Suppose $P = (A, b, c)$ is an integer minimization program in which $0 \leq a_{ij} \leq b_i$ and $c_j > 0$ for all i and j. Then $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ as $n \to \infty$.*

This is a special case of Theorem 1. The hypothesis of Theorem 2 gives us $a_{ij}/b_i \leq 1$ for all i, j, so that condition (1) is automatically satisfied by any optimum solution-vector α.

Theorem 2 is strictly weaker than Theorem 1, because condition (1) is strictly weaker than $a_{ij} \leq b_i$. For example, the set of (x, y) in the positive quadrant for which the matrix $\left(\begin{smallmatrix} x & y \\ y & x \end{smallmatrix}\right)$ satisfies (1) when $b = c = (1, 1)^T$ is the region $\{(x, y) : x^x y^y \leq 1\}$, which strictly includes the unit square. It can be shown that the condition $x^x y^y \leq 1$ is sharp for programs of this kind; that is, $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ if and only if $x^x y^y \leq 1$. It would be interesting to know if condition (1) is sharp in general.

Section 2 of this paper outlines the relationship between integer programming and hypergraph theory and gives the basic results on tensor powers of integer programs. Section 3 contains a probabilistic proof of Theorem 1. Section 4 contains a constructive (in fact, greedy) proof of Theorem 2.

2. Background and preliminary results. First, we briefly recapitulate the discussion of hypergraphs and integer programs contained in [3]. A *hypergraph* $\mathcal{H} = (V, E)$ is a finite vertex set V together with a collection $E \subset 2^V$ of nonempty subsets of V, called (hyper)edges. A *cover* of $\mathcal{H}$ is a set of vertices C that intersects every edge of $\mathcal{H}$; that is, for all $e \in E$, $C \cap e \neq \phi$. The *covering number* $\tau(\mathcal{H})$ is the smallest cardinality of a cover of $\mathcal{H}$. Suppose $\mathcal{H}$ has d vertices and m edges; then the *incidence matrix* of $\mathcal{H}$ is the m-by-d matrix A with (i,j)th entry equal to 1 if the ith edge contains the jth vertex, and equal to 0 otherwise. Furthermore, if we let b and c be vectors of length m and d, respectively, consisting entirely of 1's, and associate with each cover C of $\mathcal{H}$ a d-vector x whose jth entry is 1 or 0, according to whether or not C contains the jth vertex of $\mathcal{H}$, then $\tau(\mathcal{H})$ is seen to equal the value of the integer minimization program (A, b, c). This integer programming viewpoint naturally leads us to consider the relaxed version of the program in which the integrality constraint has been dropped; the value of the relaxed program is called the *fractional covering number* $\tau^*(\mathcal{H})$ of $\mathcal{H}$.

The definitions that appear in §1 all correspond to notions that have already been used in the theory of hypergraphs; for example, if P_i is the program that corresponds to the problem of determining $\tau(\mathcal{H}_i)$ $(i = 1, 2)$, then $P_1 \otimes P_2$ corresponds to the problem of determining $\tau(\mathcal{H}_1 \times \mathcal{H}_2)$, where $\mathcal{H}_1 \times \mathcal{H}_2$ is the hypergraph defined as follows (with $\times$ denoting Cartesian product on the right):

$$\begin{aligned} V(\mathcal{H}_1 \times \mathcal{H}_2) &= V(\mathcal{H}_1) \times V(\mathcal{H}_2), \\ E(\mathcal{H}_1 \times \mathcal{H}_2) &= \{e_1 \times e_2 : e_1 \in E(\mathcal{H}_1), e_2 \in E(\mathcal{H}_2)\}. \end{aligned}$$

In [8], McEliece and Posner proved (in different notation) a special case of Theorem 1, namely,

$$\lim_{n \to \infty} \sqrt[n]{\tau(\mathcal{H}^n)} = \tau^*(\mathcal{H}) .$$

This amounts to our Theorem 1 in the special case that the matrix A consists entirely of 0's and 1's, and the vectors b and c consist entirely of 1's. This analogy suggests the following definition.

DEFINITION 3. A program P is a *fuzzy hypergraph covering* (FHC) program if all b_i and c_j are equal to 1 and $0 \le a_{i,j} \le 1$ for all i, j. (The terminology arises by analogy with fuzzy sets.)

This paper extends McEliece and Posner's result to a more general class, including FHC programs.

Remark. It is not immediately clear that the condition $c_j = 1$ for all j is inessential, but an argument for this is given to finish the proof of Theorem 1 after it has been proved in the case where $c_j = 1$ for all j. Note, however, that the normalization of b to a vector of ones breaks the symmetry between P and $P^\perp$ and may thus change whether $v(P) = v(P^\perp)$. Finally, the condition $0 \le a_{ij} \le 1$ may not be entirely relaxed without invalidating the theorem (see the second-to-last paragraph of this section).

Example. A typical FHC program is the following: "Minimize $x_1 + x_2$ subject to

$$x_1 + \frac{1}{2}x_2 \ge 1, \tag{2}$$

$$\frac{1}{3}x_1 + x_2 \ge 1, \tag{3}$$

with $x_1, x_2 \ge 0$." This program P is associated with the matrix

$$A = \begin{pmatrix} 1 & \frac{1}{2} \\ \frac{1}{3} & 1 \end{pmatrix}.$$

Clearly, $v(P) = 2$, with optimal solution vectors $x = (1,1)$ and $(0,2)$. To determine $v^*(P)$, note that the feasibility of $x = (\frac{3}{5}, \frac{4}{5})$ implies that $v^*(P) \le \frac{7}{5}$, while the feasibility of $y = (\frac{4}{5}, \frac{3}{5})$ for the dual program $P^\perp$ implies that $v^*(P) = v^*(P^\perp) \ge \frac{7}{5}$. The tensor square of this program, $P^{\otimes 2}$, has coefficient matrix

$$A^{\otimes 2} = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{3} & 1 & \frac{1}{6} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{6} & 1 & \frac{1}{2} \\ \frac{1}{9} & \frac{1}{3} & \frac{1}{3} & 1 \end{pmatrix},$$

and we readily see that $x = (0,1,1,1)$ is a solution vector, so that $v(P^{\otimes 2}) \le 3$. This illustrates that $v(P^{\otimes 2})$ may be strictly less than $v(P)^2$. Here $v(P^{\otimes 2})^{1/2} = \sqrt{3}$ and, in fact, by Theorem 1, $v(P^{\otimes n})^{1/n} \to \frac{7}{5}$.

The following proposition does not make use of the FHC property, but the fuzzy hypergraph point of view may still be helpful to the reader in interpreting the statements and their proofs.

PROPOSITION 4. *The following hold:*

(i) $v^*(P \oplus P') = v^*(P) + v^*(P')$;
(ii) $v(P \oplus P') = v(P) + v(P')$;
(iii) $v^*(P \otimes P') = v^*(P)v^*(P')$;
(iv) $v(P \otimes P') \le v(P)v(P')$.

Proof. To prove (i) and (ii) note that if x and x' are solution vectors for P and P', then their concatenation is a solution vector for $P \oplus P'$; and, conversely, every solution vector for $P \oplus P'$ is such a concatenation. To prove (iii), suppose x and x' are optimal solution vectors to the respective linear programs P and P'. Then since

$$(A \otimes A')(x \otimes x') = (Ax) \otimes (A'x') \ge b \otimes b'$$

(note the use of nonnegativity), $x \otimes x'$ is a feasible vector for the product program $P \otimes P'$, with

$$\begin{aligned}(c \otimes c')^T(x \otimes x') &= (c^T x)(c'^T x') \\ &= v^*(P)v^*(P'),\end{aligned}$$

so that $v^*(P \otimes P') \le v^*(P)v^*(P')$. On the other hand, suppose that y and y' are optimal solution vectors to the dual programs $P^\perp$ and $P'^\perp$; then $y \otimes y'$ is a feasible vector for the program $P^\perp \otimes P'^\perp = (P \otimes P')^\perp$ with

$$\begin{aligned}(b \otimes b')^T(y \otimes y') &= (b^T y)(b'^T y') \\ &= v^*(P)v^*(P'),\end{aligned}$$

so that $v^*(P \otimes P') \ge v^*(P)v^*(P')$. (Note that we have applied the duality theorem three times: to P, to P', and to $P \otimes P'$.) We conclude that $v^*(P \otimes P') = v^*(P)v^*(P')$. The proof of (iv) is the same as the first half of the proof of (iii) (but we no longer have a duality principle to provide us with the reverse inequality). $\square$

The preceding proposition gives us an upper bound on $v(P \otimes P')$. The following less obvious result (an extension of the first inequality in Füredi's formula [3, (5.14)]) gives us a lower bound.

PROPOSITION 5. *It holds that* $v(P \otimes P') \ge \max\{v^*(P)v(P'), v(P)v^*(P')\}$.

Proof. Put $P = (A, b, c)$, $P' = (A', b', c')$. By symmetry, it is enough to show that

$$v^*(P) \leq \frac{v(P \otimes P')}{v(P')}$$

when $v(P') > 0$. Given an optimal solution vector z to $P \otimes P'$, indexed by pairs (k, l), where $1 \leq k \leq d$ and $1 \leq \ell \leq d'$, define

$$x_k = \frac{1}{v(P')} \sum_l c'_l z_{(k,l)} \, .$$

We show that x is a feasible solution to the LP-relaxation of P. Fix i and note that

$$\sum_k a_{ik} x_k = \sum_k a_{ik} \frac{1}{v(P')} \sum_l c'_l z_{(k,l)} = \frac{b_i}{v(P')} \sum_l c'_l \sum_k \frac{a_{ik}}{b_i} z_{(k,l)} \, .$$

Setting

$$y_l = \sum_k \frac{a_{ik}}{b_i} z_{(k,l)},$$

we get

$$\sum_k a_{ik} x_k = \frac{b_i}{v(P')} \sum_l c'_l y_l \, .$$

However, since

$$\sum_l a'_{jl} y_l = \frac{1}{b_i} \sum_{k,l} a_{ik} a'_{jl} z_{(k,l)} = \frac{1}{b_i} \left((A \otimes A')(z) \right)_{(i,j)} \geq \frac{1}{b_i} (b \otimes b')_{(i,j)} = b'_j$$

for all j, y is a feasible solution to P', and hence satisfies

$$\sum_l c'_l y_l \geq v(P') \, .$$

Thus

$$\sum_k a_{ik} x_k \geq \frac{b_i}{v(P')} v(P') = b_i,$$

establishing that x is indeed a feasible solution to P. We conclude that

$$v^*(P) \leq \sum_k c_k x_k = \sum_k c_k \frac{1}{v(P')} \sum_l c'_l z_{(k,l)} = \frac{1}{v(P')} \sum_{k,l} c_k c'_l z_{(k,l)} = \frac{v(P \otimes P')}{v(P')},$$

which was to be shown. □

Proposition 4(iii) implies that $\sqrt[n]{v(P^{\otimes n})} \geq \sqrt[n]{v^*(P^{\otimes n})} = v^*(P)$. Our main theorem states that if P is an FHC program or, more generally, if P satisfies condition (1), then, in fact, $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ as $n \to \infty$. Our first proof of this fact relies heavily on the ideas of McEliece and Posner and, in particular, uses the same sort of probabilistic construction as theirs did; however, our argument is necessarily more complicated, since optimal solution vectors x will typically need to have entries much larger than 1 to satisfy the constraints. In our second proof, we use a greedy construction as in Lovász's proof of the McEliece and Posner theorem [6].

It should be mentioned that the convergence $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ does not hold for integer minimization programs in general. As an illustration of this, let P be the program "Minimize $x+y+z$ subject to $2x \geq 1$, $2y \geq 1$, $2z \geq 1$ with $x, y, z \geq 0$ and integral." Then $v(P^{\otimes n}) = 3^n$ for all n, whereas $v^*(P) = \frac{3}{2}$. Hence, we see that for convergence to $v^*(P)$ to hold, something like the FHC property is required.

It should also be mentioned that the convergence $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ typically does not hold for integer maximization programs, even when all of the a_{ij} are 0's and 1's. For example, consider the problem P of maximizing $x_1+x_2+x_3+x_4+x_5$ subject to the constraints that x_1+x_2, x_2+x_3, x_3+x_4, x_4+x_5, and x_5+x_1 all be at most 1. Viewed as an integer program, this is equivalent to finding the largest independent set of vertices in the pentagon graph C_5. More generally, the nth power of P is equivalent to finding the largest independent set of vertices in the nth strong power of C_5 (see [4] for graph-product and graph-power terminology). The limit $\sqrt[n]{v(P)}$ is known as the Shannon capacity of the graph C_5 [9]. It has been shown [5] that the Shannon capacity of the graph C_5 is $\sqrt{5}$; on the other hand, $v^*(P)$ is $\frac{5}{2}$, since $(1/2, 1/2, 1/2, 1/2, 1/2)$ is a solution to both P and $P^{\perp}$. This example shows that Theorem 1 does not dualize to a theorem about maximization programs; that is, $\sqrt[n]{v((P^{\perp})^{\otimes n})}$ need not approach $v^*(P^{\perp}) = v^*(P)$.

3. Proof of Theorem 1. The proof of Theorem 1 requires some ideas from the theory of two-player zero-sum games. Treat the matrix A as the payoff matrix in a two-player zero-sum game between Alpha, who names a variable (column of A), and Beta, who names a constraint (row of A), where Alpha tries to maximize the payoff and Beta tries to minimize the payoff; the payoff is a_{ij} when constraint i and variable j are chosen. (To prepare for the multi-indices that are to follow, write a_{ij} as $a(i,j)$.) Alpha has an optimal mixed strategy that chooses each variable $x(j)$ with some probability $u(j)$. The expected value of the payoff under this strategy is called the value of the game (see [10]) and will be denoted by $\mathcal{S}$. There is a very simple relationship between this game and the LP-relaxation of the program P with matrix A and with b and c consisting of ones, namely, that if $(\alpha(1), \cdots, \alpha(d))$ is a feasible solution for the linear program, then $u(j) = \alpha(j)/\sum_j \alpha(j)$ gives a strategy for Alpha with a guaranteed payoff of at least $1/\sum_j \alpha(j)$. Moreover, $v^*(P)$ (the value of the linear program P) is equal to $1/\mathcal{S}$ (the reciprocal of the value of the matrix game), with each optimal solution-vector α giving rise to an optimal strategy u. In the case where the c vector is not all ones, it will still be convenient to let $u(j)$ denote $\alpha(j)/\sum_j \alpha(j)$, where $(\alpha(1), \cdots, \alpha(d))$ (a feasible solution with minimal cost) is given by the hypothesis of Theorem 1.

To illustrate this, consider the previous example of minimizing $x_1 + x_2$ subject to the constraints $x_1 + \frac{1}{2}x_2 \geq 1$ and $\frac{1}{3}x_1 + x_2 \geq 1$; $x_1 + x_2$ is minimized by choosing $x_1 = \frac{3}{5}$ and $x_2 = \frac{4}{5}$, and $v^*(P) = \frac{7}{5}$. The best strategy for Alpha in the game with payoff matrix

$$A = \begin{pmatrix} 1 & \frac{1}{2} \\ \frac{1}{3} & 1 \end{pmatrix}$$

is to choose $u(1) = \frac{3}{5}/\frac{7}{5} = \frac{3}{7}$ and $u(2) = \frac{4}{7}$. Then $\mathcal{S} = \frac{5}{7}$, which is clear from the fact that the expected payoff against this strategy is $\frac{5}{7}$, whether Beta chooses constraint 1 or constraint 2 or any probabilistic combination of the two. In other words, $\sum_j a(i,j)u(j) = \frac{5}{7}$ for $i = 1, 2$.

Remark. For ease of exposition, we will assume hereafter that, as in the preceding example, all of Beta's strategies are equivalent against Alpha's chosen optimal strategy; i.e.,

$$\sum_j a(i,j)u(j) = S \tag{4}$$

for all i, where $u(j) = \alpha(j)/\sum_j \alpha(j)$ with α as in the statement of Theorem 1. There is no loss of generality in doing so, since if this is not the case, there is always a way to make it be true by diminishing some of the $a(i,j)$ without changing the value of the game (in other words, without making the integer programming problem or its LP-relaxation any easier). Informally, this amounts to reining in the slack in all the constraints where the inequality is strict for the optimal solution vector.

The proof requires a probabilistic construction. Assume without loss of generality that b_i are all equal to one, since v, v^*, and the truth of condition (1) for P and its tensor powers are all preserved by the normalization that divides each a_{ij} by b_i and sets b_i equal to one. Then (1) becomes

$$\prod_j a(i,j)^{a(i,j)u(j)} \leq 1 \tag{5}$$

for all i. For ease of exposition, assume also that the c vector is all 1's (the last paragraph of the proof handles the case of general positive c vectors). Let v_0 be any constant greater than $v^*(P)$ and let $V = \lceil v_0^n \rceil$ for n large (just how large, we will decide later). It will be shown via (12) below that $v^*(P) \geq 1$ and hence that $V/v_0^n \to 1$. To determine a set of values for the d^n variables in the n-fold tensor product of P such that the sum of the variables is V, begin with all the variables equal to zero, and then select one of them according to a certain probability distribution and increment it by 1. Repeat this V times with the choices being independent and identically distributed. It will be shown that for the correct choice of probability distribution, this procedure has a positive probability (in fact, a probability close to 1) of producing a feasible integer vector. The probability distribution is exactly the same as the probability distribution used by McEliece and Posner [8]. That is to say, the probability of choosing the variable $x(j_1, j_2, \cdots, j_n)$ is given by $u(j_1)u(j_2)\cdots u(j_n)$ where u is the optimal strategy for Alpha. The proof that this construction works, however, is more involved than the one in the paper by McEliece and Posner.

Proof of Theorem 1. Choose a vector at random according to the scheme described in the previous paragraph. The random vector will be feasible if for every constraint $C(i_1, \cdots, i_n)$ the sum of the coefficients of the V randomly chosen variables in that constraint is at least 1. For each variable $x = x(j_1, \cdots, j_n)$, the coefficient in the constraint $C = C(i_1, \cdots, i_n)$ is just the product

$$\prod_{k=1}^{n} a(i_k, j_k).$$

Note that the value of this product depends on the number of times each pair (i,j) occurs in the list of (i_k, j_k), but not on the order of these pairs, and with that in mind define the *type* of the constraint-and-variable pair (C, x) to be the matrix Y, where $Y(i,j)$ is $(1/n)$ times the number of times the pair (i,j) occurs in the list of (i_k, j_k). Also, define

$$r = r(Y) = \prod_{i,j} a(i,j)^{Y(i,j)}$$

so that the coefficient of x in C is just $r(Y)^n$.

The proof will proceed by finding for each constraint C a matrix Y for which, with high probability, the number of times we select a variable x such that (C, x) is of type Y is at least $r(Y)^{-n}$. In other words, the coefficients in C of the randomly chosen variables sum to at least $r^{-n}r^n = 1$, even if we ignore all but those variables x for which (C, x) is of one particular type. (This is less surprising than it might at first seem, since the number of types is polynomial in n, whereas all other quantities are growing exponentially; hence, in restricting to those x for which (C, x) is of a certain type, we are not losing an exponentially significant contribution.)

Fix a particular constraint $C = C(i_1, \cdots, i_n)$ and define its type β to be the vector of length m such that $\beta(i)$ is equal to $(1/n)$ times the number of times i appears in the list of the i_k. Define the m-by-d matrix

$$Z(i,j) = a(i,j)\beta(i)u(j)/S. \tag{6}$$

Note that

$$\sum_j Z(i,j) = \beta(i)\left(\sum_j a(i,j)u(j)/S\right) = \beta(i), \tag{7}$$

by (4). Also, note that $r = r(Z) = \prod_{i,j} a(i,j)^{a(i,j)u(j)\beta(i)/S}$, which is the product over i of positive powers of quantities $\prod_j a(i,j)^{a(i,j)u(j)}$, each of which is at most 1 (by (5)); thus

$$r \leq 1. \tag{8}$$

Now define an approximation $\tilde{Z}$ to Z recursively in j by

$$\tilde{Z}(i,j) = \frac{1}{n}\lceil nZ(i,j)\rceil \quad \text{if } \sum_{j'=1}^{j-1} \tilde{Z}(i,j') \leq \sum_{j'=1}^{j-1} Z(i,j'), \quad \text{and}$$

$$\tilde{Z}(i,j) = \frac{1}{n}\lfloor nZ(i,j)\rfloor \quad \text{if } \sum_{j'=1}^{j-1} \tilde{Z}(i,j') > \sum_{j'=1}^{j-1} Z(i,j').$$

The important properties of $\tilde{Z}$ are that

(i) $Z(i,j) = 0$ implies $\tilde{Z}(i,j) = 0$;
(ii) $n\tilde{Z}(i,j)$ is an integer;
(iii) $|Z(i,j) - \tilde{Z}(i,j)| < 1/n$; and
(iv) $\sum_j \tilde{Z}(i,j) = \sum_j Z(i,j) = \beta(i)$.

They follow immediately from the definition. The reason we want conditions (i) and (iii) to hold is so that calculations involving $\tilde{Z}$ can be approximated by calculations involving Z; the reason we want conditions (ii) and (iv) is so that $\tilde{Z}$ will actually be the type of a constraint-and-variable pair (C, x) for some x.

Define

$$\tilde{r} = \tilde{r}(Z) = \prod_{i,j} a(i,j)^{\tilde{Z}(i,j)}.$$

The immediate object is to estimate the number of variables x of the V that are chosen (with repetition) for which the pair (C, x) is of the type $\tilde{Z}$, and show that this number is very likely to be at least $\tilde{r}^{-n}$. Each time a variable $x = x(j_1, \cdots, j_n)$ is chosen, the chance that (C, x) is of the type $\tilde{Z}$ is just the chance that for each i, the values of the j_k for which $i_k = i$ form the multiset that has $n\tilde{Z}(i,1)$ ones, $n\tilde{Z}(i,2)$ twos, and so on. Denote this probability by $P(\tilde{Z})$. Then

$$P(\tilde{Z}) = \prod_i \left[\text{multi}\left(n\beta(i); n\tilde{Z}(i,1), \cdots, n\tilde{Z}(i,m)\right) \prod_j u(j)^{n\tilde{Z}(i,j)} \right],$$

where multi$(x; y_1, y_2, \cdots)$ denotes the multinomial coefficient with x on top and $y_1, y_2, \cdots$ on the bottom. Evaluate these multinomial coefficients by assuming $n \geq 3$ and by using the inequalities

$$x! > x^x e^{-x} \quad \text{and} \quad x! < n x^x e^{-x}$$

for the numerator and denominator, respectively. Here $0^0 = 1$ by convention. After all the e^x and n^x factors cancel, we obtain

$$P(\tilde{Z}) \geq n^{-md} \left[\prod_i \beta(i)^{\beta(i)} \prod_{i,j} \tilde{Z}(i,j)^{-\tilde{Z}(i,j)} \prod_{i,j} u(j)^{\tilde{Z}(i,j)} \right]^n . \tag{9}$$

Note that if we replace $\tilde{Z}$ by Z everywhere in the bracketed expression, it becomes

$$\begin{aligned} &\prod_i \beta(i)^{\beta(i)} \prod_{i,j} Z(i,j)^{-Z(i,j)} \prod_{i,j} u(j)^{Z(i,j)} \\ &= \prod_{i,j} \left(u(j)\beta(i)Z(i,j)^{-1}\right)^{Z(i,j)} \\ &= \prod_{i,j} \left(\mathcal{S}a(i,j)^{-1}\right)^{Z(i,j)} \qquad (10) \\ &= \mathcal{S} \prod_{i,j} a(i,j)^{-Z(i,j)} \\ &= \mathcal{S}/r, \end{aligned}$$

where the first equality follows from (7) and the second follows from the definition of Z in (6). We proceed to rewrite (9) in terms of $\mathcal{S}/r$. Specifically, we will approximate (9) by a version with Z replacing $\tilde{Z}$, thereby introducing an additional error factor of the form $(1-\delta(n))^n$ with $\delta(n) \to 0$ as $n \to \infty$. By property (iii) of $\tilde{Z}$,

$$\frac{\tilde{Z}(i,j)^{-\tilde{Z}(i,j)}}{Z(i,j)^{-Z(i,j)}} \geq \inf \frac{x^x}{y^y},$$

where the infimum is taken over nonnegative x and y satisfying $|x - y| < 1/n$. Denote this infimum by $1 - \theta(n)$; since the function $x \ln(x)$ (with $0 \ln 0$ defined to be 0) is uniformly continuous on $[0,1]$, $\theta(n) \to 0$ as $n \to \infty$. Thus, putting $1 - \delta_1(n) = (1 - \theta(n))^{md}$, we get

$$\frac{\prod_{i,j} \tilde{Z}(i,j)^{-\tilde{Z}(i,j)}}{\prod_{i,j} Z(i,j)^{-Z(i,j)}} \geq (1-\theta(n))^{md} = 1 - \delta_1(n)$$

with $\delta_1(n) \to 0$ as $n \to \infty$. Also note that

$$\frac{u(j)^{\tilde{Z}(i,j)}}{u(j)^{Z(i,j)}} = u(j)^{\tilde{Z}(i,j)-Z(i,j)},$$

which is at least $u(j)^{1/n}$ when $u(j) > 0$ and is 1 when $u(j) = 0$; either way, the fraction is at least ${u_{\min}}^{1/n}$, where $u_{\min}$ is the minimum of the positive entries of u. Thus

$$\frac{\prod_{i,j} u(j)^{\tilde{Z}(i,j)}}{\prod_{i,j} u(j)^{Z(i,j)}} \geq \left({u_{\min}}^{1/n}\right)^{md}.$$

Letting $\delta_2(n)$ be defined by $1 - \delta_2(n) = {u_{\min}}^{md/n}(1 - \delta_1(n))$, we conclude that

$$\begin{aligned} P(\tilde{Z}) &\geq n^{-md}(1-\delta_2(n))^n \left[\prod_i \beta(i)^{\beta(i)} \prod_{i,j} Z(i,j)^{-Z(i,j)} \prod_{i,j} u(j)^{Z(i,j)}\right]^n \\ &= n^{-md}(1-\delta_2(n))^n (\mathcal{S}/r)^n , \end{aligned} \tag{11}$$

where $\delta_2(n) \to 0$ as $n \to \infty$.

The other estimate of this sort that we will need is a bound on $\tilde{r}$ in terms of r. Take $\eta > 0$ with $\eta \leq a(i,j)$ and $\eta \leq 1/a(i,j)$ for all $a(i,j) \neq 0$. Then

$$\frac{a(i,j)^{\tilde{Z}(i,j)}}{a(i,j)^{Z(i,j)}} = a(i,j)^{\tilde{Z}(i,j)-Z(i,j)} \geq \eta^{1/n}$$

for all i, j, and

$$\begin{aligned} \tilde{r} &= \prod_{i,j} a(i,j)^{\tilde{Z}(i,j)} \\ &\geq \prod_{i,j} \eta^{1/n} \prod_{i,j} a(i,j)^{Z(i,j)} \\ &= \eta^{md/n}\, r . \end{aligned}$$

Then from (11) it follows that

$$P(\tilde{Z}) \geq Q(n)(\mathcal{S}/\tilde{r})^n, \tag{12}$$

where $Q(n) = (1-\delta_2(n))^n \eta^{md} n^{-md}$. (Note that $\theta, \delta_1, \delta_2$, and Q depend only on n, not on which constraint was chosen. What will end up being important about $Q(n)$ is that $\sqrt[n]{Q(n)} \to 1$ since $\delta(n) \to 0$.) Since $\mathcal{S}/r = \lim_n \mathcal{S}/\tilde{r} \leq \lim_n Q(n)^{1/n} P(\tilde{Z})^{1/n} \leq \lim_n Q(n)^{1/n} = 1$, it follows that $v^*(P) = 1/\mathcal{S} \geq 1/r \geq 1$, and the debt we incurred in the paragraph before the proof of Theorem 1 by claiming $V/v_0^n \to 1$ is paid. A more meaningful interpretation of (12) is that if we choose a variable $x = x(j_1, \cdots j_n)$ at random with probability $u(j_1) \cdots u(j_n)$, the probability that (C, x) is of type $\tilde{Z}$ is at least $Q(n)(\mathcal{S}/\tilde{r})^n$. Also, recall that if (C, x) is of type $\tilde{Z}$, then the coefficient of x in C is $\tilde{r}$.

The last step in the proof of Theorem 1 is an application of the following lemma.

LEMMA 6. *Let a, b, c, ϵ be positive real numbers with $ab/(1+\epsilon) \geq c \geq 1 \geq b$. Consider a family $\{X_i\}$ of at least a^n independently and identically distributed Bernoulli random variables with $\mathbf{P}(X_i = 1) \geq b^n$. Then there is some positive constant δ and some positive integer N for which $\mathbf{P}(\sum X_i < c^n) < e^{-e^{\delta n}}$ whenever $n > N$. Furthermore, N and δ can be chosen to depend only on ϵ.*

Assuming the lemma for the moment, the rest of the proof of Theorem 1 is as follows. We have selected $V = \lceil v_0{}^n \rceil$ variables for some $v_0 > v^*(P) = 1/S$. Then for any fixed constraint $C = C(i_1, \cdots, i_n)$, we have chosen a matrix $\tilde{Z}$ and its associated value $\tilde{r}$ so that each variable chosen by our random scheme has coefficient at least $\tilde{r}^n$ with probability at least $P(\tilde{Z})$. Let $a = v_0$, $b = P(\tilde{Z})^{1/n}$, and $c = \max(1, 1/\tilde{r})$. Let X_i be the Bernoulli random variable that equals 1 if the ith variable chosen, x, has the property that (C, x) is of type $\tilde{Z}$ and equals 0 otherwise. Since $\tilde{r}$ converges to r as n gets large, and since $r \leq 1$ by (8), it follows that for any $\delta > 0$, $1/\tilde{r} > 1 - \delta$ for sufficiently large n. Then (12) implies that the first inequality in the hypothesis of Lemma 6 is satisfied with any $\epsilon < v_0 S - 1$ for sufficiently large n, since $ab/c \geq v_0 S Q(n)^{1/n}/c\tilde{r}$ and $Q(n)^{1/n}/c\tilde{r} \to 1$. The second inequality is guaranteed by the choice of c and the last is true because b is a positive power of a probability. The conclusion of the lemma is that the probability of there being enough variables of type $\tilde{Z}$ to satisfy the constraint (namely, $\tilde{r}^{-n}$ of them) is at least $1 - e^{-e^{\delta n}}$. This is true uniformly over all constraint types for sufficiently large n, and since there are only exponentially many constraints C, the sum of the failure probabilities over all constraints goes to zero as n goes to infinity. In particular, the constraints are all satisfied with nonzero probability for n sufficiently large, and that proves the theorem.

The case where the c vector is not all 1*'s.* Suppose that $(\alpha(1), \cdots, \alpha(d))$ is an optimal solution to the program. Then letting $u(j) = \alpha(j)/\sum_j \alpha(j)$ gives a strategy in the two-player game that achieves a payoff of $1/\sum_j \alpha(j)$. Letting $S = 1/\sum_j \alpha(j)$, the calculation after (7) still shows that condition (1) implies $r \leq 1$, and (12) still gives $S \leq 1$. Here we have borrowed another page from matrix-game theory to assert that the optimal solution with the new c vector will, in general, have a different set of dominated strategies for Alpha (i.e., a different set of j for which $u(j) = 0$), but that each $\sum_j a(i,j)u(j)$ will still be $1/S$ for all i such that strategy i is not a dominated strategy for Beta. Note that since the set of dominated strategies changes with the c vector, the modification of dominated strategies as in the paragraph before condition (1) must come after looking at the c vector. Ignore the cost vector c for the moment and use the same randomized algorithm as before to choose $\lceil (\epsilon + \sum_j \alpha(j))^n \rceil$ variables to increment, where ϵ is a new, arbitrarily small, positive number. Since the number of variables is going to infinity, the constraints are now satisfied with a probability that goes to one as $n \to \infty$. The expected total cost of the variables chosen is $\lceil (\epsilon + \sum_j \alpha(j))^n \rceil$, so the probability that the cost exceeds $(2\epsilon + \sum_j \alpha(j))^n (c^T u)^n$ goes to zero as n goes to infinity; hence, in particular, the probability that the cost is at most $(2\epsilon + \sum_j \alpha(j))^n (c^T u)^n$, and that the constraints are all satisfied that, is positive for large n. However, $c^T u = c^T \alpha / \sum_j \alpha(j)$, so, for large enough n, there are feasible integer vectors with cost at most $((2c^T\alpha/\sum_j \alpha_j)\epsilon + c^T\alpha)^n$ for arbitrarily small ϵ. The theorem is proved. $\square$

Proof of Lemma 6. This is a standard large deviation estimate, but, to get δ to depend only on ϵ, the usual moment estimate will be redone from scratch. The following fact can easily be seen by looking at chords of the graph of $\ln(1-x)$ near

$x = 0$, below:

$$\frac{\ln(1-zu)}{z\ln(1-u)} \to 1 \text{ uniformly over } z \in [0,1] \text{ as } u \downarrow 0. \tag{13}$$

(The expression is taken to be 1 when z or u is 0.) Letting $t \geq 0$ be a free parameter, the moment calculation is

$$\begin{aligned}
\mathbf{P}(\textstyle\sum X_i < c^n) &= \mathbf{P}(e^{t\sum -X_i} > e^{-tc^n}) \\
&\leq \mathbf{E}e^{t(-\sum X_i)} \,/\, e^{-tc^n} \\
&\leq (\mathbf{E}e^{-tX_1})^{a^n}/e^{-tc^n} \\
&\leq (1-b^n(1-e^{-t}))^{a^n}/e^{-tc^n}.
\end{aligned}$$

Exponentiating (13), we see that for all $\gamma \in (0,1)$, u can be chosen small enough so that for any $z \in [0,1]$, $1 - zu \leq ((1-u)^z)^\gamma$. Then with $z = b^n$ and $u = 1 - e^{-t}$ we have that for any $\gamma \in (0,1)$ and sufficiently small t, the following inequality holds for any $b \in [0,1]$:

$$1 - b^n(1-e^{-t}) \leq \left([1-(1-e^{-t})]^{b^n}\right)^\gamma = e^{-t\gamma b^n}.$$

Then

$$\mathbf{P}(\textstyle\sum X_i < c^n) \leq e^{-t(\gamma a^n b^n - c^n)}. \tag{14}$$

Fix any γ such that $\gamma ab > c$. The right-hand side of (14) increases when b and c are decreased by the same factor, and also when b is decreased, so assume without loss of generality that $ab/(1+\epsilon) = c = 1$. Then the exponent in the right-hand side grows like $(1+\epsilon)^n$, so for any $\delta \in (0, \ln(1+\epsilon))$, there is an N for which the left-hand side of (14) is bounded by $e^{-e^{n\delta}}$ whenever $n > N$. It is clear that δ and N can be chosen to depend only on ϵ. □

4. Proof of Theorem 2. The proof of Theorem 1 made delicate use of the structure of the nth power of an integer program. In contrast, the proof presented in this section is based on very general lemmas about semi-FHC programs (defined below), and only at the very end does the notion of a product of integer programs make an appearance. Even then, we appeal only to the most basic facts about $P^{\otimes n}$— namely, that $v^*(P^{\otimes n}) = v^*(P)^n$, and that the number of constraints in $P^{\otimes n}$ grows exponentially, not faster.

We will prove the following restatement of Theorem 2 (obtained by rescaling, as in the proof of Theorem 1).

THEOREM 7. *Suppose that $P = (A, b, c)$ is an integer minimization program in which $0 \leq a_{ij} \leq 1$, $b_i = 1$, and $c_j > 0$ for all i and j. Then $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$ as $n \to \infty$.*

We will prove Theorem 7 by analyzing a somewhat broader class of programs than those that satisfy its hypothesis. Say that an integer program $P = (A, b, c)$ is of *semi-FHC type* if all of the entries of A are in $[0,1]$ and all of the entries of c are positive (the entries of b may be arbitrary real numbers). Call a constraint trivial if it is satisfied by all real vectors x (i.e., all its coefficients are 0 and its right-hand side is less than or equal to 0), and call a program trivial if all its constraints are trivial. Given an integer program $P = (A, b, c)$ of semi-FHC type, let m denote the number of nontrivial constraints, $S(P)$ denote the sum of the entries of b, and $D(P)$ denote $\max_j\{\sum_i a_{ij}/c_j\}$. When P is of FHC type, $S(P) = m$; when P is trivial, $S(P) \leq 0$. (The notation "$D(P)$" originates from the fact that when P is a hypergraph-covering program, $D(P)$ coincides with the maximum degree of the hypergraph, i.e., the maximum number of edges sharing a vertex.)

Our argument begins with the observation that $v^*(P) \geq S(P)/D(P)$ for any nontrivial semi-FHC program P. To see this, let y be the vector of length m, all of whose components equal $1/D(P)$. Since the jth row-sum of A^T is at most $D(P)c_j$, the jth component of the vector $A^T y$ is less than or equal to c_j for every j. Hence y is a feasible solution to the dual program $P^\perp$, whence $v^*(P) = v^*(P^\perp) \geq b^T y = S(P)/D(P)$.

In particular, suppose that P is as in Theorem 7, and Q is a nontrivial semi-FHC program such that every feasible solution to P is also feasible for Q. Then $S(Q)/D(Q) \leq v^*(Q) \leq v^*(P)$. This upper bound on the ratio $S(Q)/D(Q)$ is the key ingredient in the proof of the following fact.

LEMMA 8. *If P is as in Theorem 7 and $c_j \leq v^*(P)$ for all j, then there is a nonnegative integer vector x^* such that $c^T x^* \leq 2(\ln 10 + 1)v^*(P)$ and at least one-fourth of the entries of the column vector Ax^* exceed* 1.

Proof. We define a sequence of semi-FHC programs $P^{(0)} = P$, $P^{(1)}$, $P^{(2)}$, $\cdots$, $P^{(N)}$, where N will be specified later, together with a sequence of d-component integer vectors $u^{(0)} = 0$, $u^{(1)}$, $u^{(2)}$, $\cdots$, $u^{(N)}$ in the following iterative way. We assume that $P^{(k)} = (A^{(k)}, b^{(k)}, c)$ has already been defined, and wish to define $P^{(k+1)}$. Take j (more properly speaking, j_k) such that $\sum_i a_{ij}^{(k)}/c_j = D(P^{(k)})$, and let $u^{(k+1)}$ be the vector obtained from $u^{(k)}$ by incrementing its jth component by 1. Let $b^{(k+1)}$ equal $b^{(k)}$ minus the jth column of $A^{(k)}$. Lastly, to define $A^{(k+1)}$, call a row of $A^{(k)}$ *satisfied* if the corresponding entry of $b^{(k+1)}$ is negative; replace all the entries in all the satisfied rows of $A^{(k)}$ by 0's and call the resulting matrix $A^{(k+1)}$. Terminate this greedy procedure after N steps, where N is chosen to be the smallest integer such that $\sum_{k=0}^{N-1} c_{j_k} \geq v^*(P) \ln 10$.

Note that under this scheme, if we fix i between 1 and m and look at the ith entries of the successive vectors $b^{(0)}, b^{(1)}, b^{(2)}, \cdots, b^{(N)}$, we see a sequence of numbers that decreases by at most 1 at each stage until a negative term appears, at which point the sequence is constant (since the corresponding row of A gets "zeroed out"). Hence all the entries of all the b-vectors lie in the interval $[-1, 1]$.

Also note that a feasible solution for $P^{(k)}$ remains feasible for $P^{(k+1)}$, since the only change made in passing from the former to the latter is that certain constraints have been relaxed (some of the entries in the b-vector have decreased), while other constraints have been effectively dropped (some of the rows of the A-matrix have been zeroed out). Therefore any feasible solution for $P^{(0)} = P$ is feasible for each $P^{(k)}$. If $P^{(k)}$ is nontrivial, this implies that $S(P^{(k)})/D(P^{(k)}) \leq v^*(P)$ and hence

$$
\begin{aligned}
S(P^{(k+1)})/S(P^{(k)}) &= \frac{S(P^{(k)}) - c_{j_k} D(P^{(k)})}{S(P^{(k)})} \\
&= 1 - c_{j_k} \frac{D(P^{(k)})}{S(P^{(k)})}
\end{aligned}
$$

$$\leq 1 - \frac{c_{j_k}}{v^*(P)}$$

$$\leq e^{-c_{j_k}/v^*(P)}$$

for all k between 0 and $N-1$. Multiplying these N inequalities together, we obtain

$$S(P^{(N)})/S(P) \leq e^{-\sum_{k=0}^{N-1} c_{j_k}/v^*(P)}$$

$$\leq e^{-\ln 10}$$

$$= \frac{1}{10} .$$

If any of the $P^{(k)}$ are trivial then so is $P^{(N)}$, so that $S(P^{(N)}) = 0$, from which we see that the foregoing inequality holds in this case as well. We have shown that the sum of the entries of $b^{(N)}$ (all of which lie between -1 and 1) is at most $\frac{1}{10}S(P) = \frac{1}{10}\sum_i b_i^{(0)} = m/10$. This means that at least a quarter of its entries are less than $\frac{1}{2}$ (since, otherwise, the average of the entries of $b^{(N)}$ would be at least $\frac{3}{4}\left(\frac{1}{2}\right) + \frac{1}{4}(-1) = \frac{1}{8} > \frac{1}{10}$, a contradiction). On the other hand, we also know that all of the entries of $b^{(0)}$ were 1's, so at least a quarter of the entries of $b^{(0)} - b^{(N)}$ must exceed $\frac{1}{2}$; since $b^{(N)} \geq b^{(0)} - Au^{(N)}$, we have $Au^{(N)} \geq b^{(0)} - b^{(N)}$, so that at least a quarter of the entries of $Au^{(N)}$ exceed $\frac{1}{2}$. Also, by the minimality of N and our assumption on c, we conclude that $c^T u^{(N)} = \sum_{k=0}^{N-1} c_{j_k} + c_{j_N} \leq v^*(P)\ln 10 + v^*(P) = (\ln 10 + 1)v^*(P)$. Setting $x^* = 2u^{(N)}$, we obtain a vector with the desired properties. □

LEMMA 9. *If P is as in Theorem 7, $S(P) > 1$, and $c_j \leq v^*(P)$ for all j, then*

$$v(P) \leq 100\, v^*(P) \ln S(P) .$$

Proof. Let x^* be the vector of Lemma 8 with $c^T x^* \leq 2(\ln 10 + 1)v^*(P)$ and with the property that at least one-fourth of the entries of Ax^* exceed 1. Let P' be the integer program obtained from P by dropping all the constraints that correspond to the values of i for which $(Ax^*)_i \geq 1$. Note that any feasible solution to P', if increased by adding x^*, yields a feasible solution to P; hence

$$v(P) \leq v(P') + 2(\ln 10 + 1)v^*(P) .$$

Note that P' has at most three-fourths as many constraints as P. Hence, iterating this reduction process K times, where

$$K = \lceil \log_{4/3} S(P) \rceil > \log_{4/3} S(P) ,$$

we obtain a program Q such that

$$v(P) \leq v(Q) + 2K(\ln 10 + 1)v^*(P) .$$

The number of constraints in Q, however, is at most

$$\left(\tfrac{3}{4}\right)^K S(P) \ < \ \left(\tfrac{3}{4}\right)^{\log_{4/3} S(P)} S(P) \ = \ 1;$$

that is, Q is the empty program, with no constraints and with value 0. Hence

$$v(P) \leq 2K(\ln 10 + 1)v^*(P)$$

$$= 2\left\lceil \frac{\ln S(P)}{\ln 4/3} \right\rceil (\ln 10 + 1) v^*(P)$$

$$\leq 100\, v^*(P)\, \ln S(P),$$

as claimed, the last inequality following from the fact that $\lceil x \rceil / x$ cannot be too large when x is bounded away from zero. □

The following lemma will permit us to assume that in Theorem 7 no component of the vector c is greater than $v^*(P)$.

LEMMA 10. *If P is as in Theorem 7 and $c_k > v^*(P)$, then every optimal solution x^* to the LP-relaxation of P has $x_k^* = 0$.*

Proof. If x^* is a feasible solution for P then so is z^*, given by

$$z_j^* = \begin{cases} x_j^*(1 + x_k^*) & \text{for } j \neq k, \\ x_k^{*2} & \text{for } j = k, \end{cases}$$

since

$$\sum_j a_{ij} z_j^* = (1 + x_k^*) \sum_j a_{ij} x_j^* - a_{ik} x_k^*$$

$$\geq (1 + x_k^*) - x_k^* = 1.$$

If x^* is optimal as well, then

$$\begin{aligned} v^*(P) &= \sum_j c_j x_j^* \\ &\leq \sum_j c_j z_j^* \\ &= \sum_{j \neq k} c_j x_j^*(1 + x_k^*) \; + \; c_k x_k^{*2} \\ &= (1 + x_k^*) \sum_j c_j x_j^* \; + \; -c_k x_k^* \\ &= (1 + x_k^*) v^*(P) - c_k x_k^* \\ &= v^*(P) - (c_k - v^*(P)) x_k^*, \end{aligned}$$

which implies that $x_k^* = 0$. □

Proof of Theorem 7. Let m be the number of constraints of P; we may suppose that $m > 1$ since the result is trivial for $m = 1$. Note that $P^{\otimes n}$ has only m^n constraints. Assume first that $c_j \leq v^*(P)$ for all j. Then all of the entries of $c^{\otimes n}$ are at most $v^*(P)^n = v^*(P^{\otimes n})$ for all n, so that by Lemma 9 we have

$$v(P^{\otimes n}) \leq 100 v^*(P^{\otimes n}) \ln(m^n) = 100 n \ln(m) v^*(P)^n,$$

from which we can conclude that $\sqrt[n]{v(P^{\otimes n})} \to v^*(P)$. On the other hand, if $c_k > v^*(P)$ for some k, then we may drop those variables x_k from the program P. This can only make $v(P^{\otimes n})$ larger, but does not affect $v^*(P)$ because of Lemma 10, so we have reduced the problem to the first case, which has just been solved. □

Acknowledgments. We would like to thank both referees for their detailed reports, which suggested many useful changes. In particular, Mark Hartmann explained to us how to modify our original proof of Theorem 7 to remove the unnecessary restriction that $c_j = 1$ for all j.

REFERENCES

[1] W. BEIN AND P. BRUCKER, *Greedy concepts for network flow problems*, Discrete Appl. Math., 15 (1986), pp. 135–144.
[2] F. M. FEKETE, *Über die Verteilung der Wurzeln bei gewissen algebraischen Gleichungen mit ganzzahligen Koeffizienten*, Math. Z., 17 (1923), pp. 228–249.
[3] Z. FÜREDI, *Matchings and covers in hypergraphs*, Graphs Combin., 4 (1988), pp. 115–206.
[4] P. HELL AND F. ROBERTS, *Analogues of the Shannon capacity of a graph*, Ann. Discrete Math., 12 (1982), pp. 155–162.
[5] L. LOVÁSZ, *On the Shannon capacity of a graph*, IEEE Trans. Inform. Theory, 25 (1979), pp. 1–7.
[6] ———, *On the ratio of optimal integral and fractional covers*, Discrete Math., 13 (1975), pp. 383–390.
[7] R. MCELIECE, *The Theory of Information and Coding*, Addison–Wesley, Reading, MA, 1977.
[8] R. MCELIECE AND E. POSNER, *Hide and seek, data storage, and entropy*, Ann. Math. Statist., 42 (1971), pp. 1706–1716.
[9] C. E. SHANNON, *The zero-error capacity of a noisy channel*, IRE Trans. Inform. Theory, IT-2 (1956), pp. 8–19.
[10] J. VON NEUMANN AND O. MORGENSTERN, *Theory of games and economic decisions*, 3rd ed., Princeton University Press, Princeton, NJ, 1953.