Journal of Global Optimization **11:** 325–338, 1997.

Global Escape Strategies for Maximizing Quadratic Forms over a Simplex

IMMANUEL M. BOMZE
Department of Statistics, Operations Research and Computer Science, University of Vienna, Vienna, Austria

(Received: 18 October 1996; accepted: 23 March 1997)

Abstract. Consider the problem of maximizing a quadratic form over the standard simplex. Problems of this type occur, e.g., in the search for the maximum (weighted) clique in an undirected graph. In this paper, copositivity-based escape procedures from inefficient local solutions are rephrased into lower-dimensional subproblems which are again of the same type. As a result, an algorithm is obtained which tries to exploit favourable data constellations in a systematic way, and to avoid the worst-case behaviour of such NP-hard problems whenever possible. First results on finding large cliques in DIMACS benchmark graphs are encouraging.

Key words: Copositivity, block pivoting, indefinite quadratic programming, maximum clique, independent set.

1. Introduction

In this paper, we condider the following optimization problem:

$$x'Ax \to \max ! \quad \text{subject to } x \in \Delta^n \,, \tag{1.1}$$

where A is an arbitrary symmetric $n \times n$ matrix; a $'$ denotes transposition; and Δ^n is the standard $(n-1)$-dimensional simplex in n-dimensional Euclidean space $\mathbb{R}^n$, i.e. the intersection of an affine hyperplane with the positive orthant $\mathbb{R}^n_+$:

$$\Delta^n = \{x \in \mathbb{R}^n_+ : e'x = 1\} \,,$$

(of course, the region $\{y \in \mathbb{R}^{n-1}_+ : e'y \leq 1\}$ can always be represented by Δ^n introducing a slack variable). Here and in the sequel, the letter e is reserved for a vector of appropriate length, consisting of unit entries exclusively. While $I = \text{diag}(e)$ denotes a generic identity matrix, we use the letters o and O (not 0) to designate zero vectors and zero matrices of suitable size, to improve readability. We also abbreviate $\mathcal{V} = \{1, \ldots, n\}$ and denote by $\#\mathcal{A}$ the size, i.e. the number of elements of a finite set $\mathcal{A}$.

Note that the maximizers of (1.1) remain the same if A is replaced with $A + \gamma ee'$ where γ is an arbitrary constant. So without loss of generality assume henceforth that all entries of A are non-negative.

Of course, quadratic optimization problems like (1.1) – even the detection of their local solutions – are NP-hard [11], [12]. Nevertheless, there are several exact procedures which try to exploit favourable data constellations in a systematic way, to avoid the worst-case behaviour whenever possible. As a prototypical example for this type of algorithms, consider the iterative procedure proposed in [2], which consists of two parts. At first, a local solution of (1.1) is generated by following the paths of feasible points provided by a dynamical system borrowed from evolutionary modelling; in the second step, the procedure escapes from an inefficient local maximizer in a way such that improvement in the objective is guaranteed. In this paper we concentrate on how to improve efficiency of the second part.

The paper is organized as follows. The proposed procedure for the general case is presented in Sections 2 and 3. Section 4 specifies an anti-greedy algorithm to obtain good pivoting blocks and gives the complete algorithm. Section 5 addresses an important application, namely the determination of a maximum clique in an undirected graph, and gives preliminary experimental results on some DIMACS benchmark graphs. See [5] for a forthcoming detailed empirical study of this approach.

2. Escaping from an Inefficient Local Solution

For ease of reference, we first repeat here the following characterization of global optimality from [2, Theorem 6]:

THEOREM 1. *Suppose that $x \in \Delta^n$ is a local solution to (1.1), and denote by $S = \{i \in \mathcal{V} : x_i > 0\}$ the set of its positive co-ordinates. Then x is a global solution of (1.1) if and only if for all $i \in S$, the $n \times n$-matrix*

$$Q_i = e_i(Ax)' + (Ax)e_i' - x_i A \tag{2.1}$$

is copositive with respect to the polyhedral cone

$$\Gamma_i = \left\{v \in \mathbb{R}^n : e'v = 0, v_r \geq 0 \text{ if } r \notin S \text{ and } \frac{v_i}{x_i} \leq \frac{v_j}{x_j} \text{ for all } j \in S\right\}, \tag{2.2}$$

which means that

$$v'Q_i v \geq 0 \quad \text{for all} \quad v \in \Gamma_i\,. \tag{2.3}$$

If there is a direction $v \in \Gamma_i$ such that $v'Q_i v < 0$, then $v_i < 0$ and

$$\tilde{x} = x - \frac{x_i}{v_i} v \in \Delta^n$$

is a strictly improving feasible point.

The next step is to decompose the copositivity condition with respect to the cones Γ_i into several subproblems of dimension $n - m < n$ by means of block pivoting as introduced for detection of copositivity in [1]. This yields a series of auxiliary problems of considerably smaller dimension. While there are quite many of these in general, the particularly simple structure of the cones Γ_i here guarantees that there are at most m such subproblems, due to the fact that the feasible set is a simplex here. Apart from generating smaller subproblems, the focus here is the deliberate detection of an improving feasible direction using the information obtained so far, rather than to restart blindly from scratch. For determining a maximum clique, this approach is followed in [2] where a block pivoting escape procedure based on the parallel generation of large cliques and independent sets is proposed. So we start from this as a vantage point and rephrase the resulting copositivity conditions as a series of smaller problems of the same type. Whenever one of these yields a (local) solution exceeding the current best value of the master problem (1.1), then an improving feasible point can be generated immediately.

To reach this goal, we have first to choose an m-element subset $T \subset \mathcal{V}$ disjoint from S such that the corresponding symmetric $m \times m$-submatrix A_T of A is copositive on $e^\perp$, the hyperplane in $\mathbb{R}^m$ where the sum of all co-ordinates vanishes. This property is equivalent to positive semidefiniteness of PA_TP, where $P = I - \frac{1}{m}ee'$ is the orthoprojector onto $e^\perp$. See Section 4 for a simple anti-greedy approach to find such a set T. Partition also

$$Q_i = \begin{bmatrix} A_i & B_i \\ B_i' & C_i \end{bmatrix}$$

according to T and $\mathcal{V} \setminus T$. Since S and T are disjoint and $i \in S$, relation (2.1) yields $A_i = -x_i A_T$. Next define the symmetric matrices of order $n - m$

$$Q_{i,j}^{(T)} = C_i - er_j' - r_j e' - (a_{jj}x_i)ee' \,, \quad j \in T \,, \tag{2.4}$$

where r_j' is the j-th row of B_i, as well as the following polyhedral cones in $\mathbb{R}^{n-m}$:

$$\Gamma_i^{(T)} = \{z \in \mathbb{R}^{\mathcal{V}\setminus T} : e'z \le 0,\, z_\ell \ge 0 \text{ if } \ell \notin S,\, \frac{z_i}{x_i} \le \frac{z_j}{x_j} \text{ if } j \in S\}\,, \tag{2.5}$$

and m subcones of this, abbreviating by $\gamma_{sj} = a_{sj} - a_{jj}$:

$$\Gamma_{i,j}^{(T)} = \{z \in \Gamma_i^{(T)} : r_s'z \ge r_j'z - (x_i\gamma_{sj})e'z \text{ for all } s \in T\}\,, \quad j \in T\,. \tag{2.6}$$

We need the following key decomposition and comparison result:

LEMMA 2. *If* $(I - \frac{1}{m}ee')A_T(I - \frac{1}{m}ee')$ *is positive semidefinite, then*

$$\bigcup_{j\in S} \Gamma_{i,j}^{(T)} = \Gamma_i^{(T)}\,.$$

To be more precise, for any $z \in \Gamma_i^{(T)}$ *choose* $j \in T$ *such that*

$$r_j'z + \tfrac{1}{2}(a_{jj}x_i)e'z \le r_s'z + \tfrac{1}{2}(a_{ss}x_i)e'z \quad \textit{for all } s \in T\,. \tag{2.7}$$

Then $z \in \Gamma_{i,j}^{(T)}$ *and, furthermore,*

$$z'Q_{i,j}^{(T)}z \le z'Q_{i,s}^{(T)}z \quad \textit{for all } s \in T\,. \tag{2.8}$$

Proof. Put $v = e_s - e_j \in e^\perp$. Then for all $s \in T$ we have

$$\begin{aligned}(r_s - r_j)'z + (x_i\gamma_{sj})e'z &= -x_i(e'z)[\tfrac{1}{2}a_{ss} - \gamma_{sj} - \tfrac{1}{2}a_{jj}]\\ &= -x_i(e'z)[\tfrac{1}{2}a_{ss} - a_{sj} + \tfrac{1}{2}a_{jj}]\\ &= -x_i\frac{e'z}{2}v'A_T v\\ &= -x_i\frac{e'z}{2}v'(I - \frac{1}{m}ee')A_T(I - \frac{1}{m}ee')v \geq 0\end{aligned}$$

by virtue of $e'z \leq 0$, which shows $z \in \Gamma_{i,j}^{(T)}$. Finally, by (2.4) and (2.7) we get

$$z'Q_{i,j}^{(T)}z - z'Q_{i,s}^{(T)}z = -2(e'z)[r_j'z + \tfrac{1}{2}(a_{jj}x_i)e'z - r_s'z - \tfrac{1}{2}(a_{ss}x_i)e'z] \leq 0\,,$$

which proves the assertion. □

THEOREM 3. *Suppose that $x \in \Delta^n$ is a local solution to (1.1), and denote by $S = \{i \in \mathcal{V} : x_i > 0\}$ the set of its positive co-ordinates, and by $k = \#S$. Pick a disjoint subset $T \subset \mathcal{V}\setminus S$ of size $\#T = m \leq n-k$ such that $(I-\frac{1}{m}ee')A_T(I-\frac{1}{m}ee')$ is positive semidefinite. Then x is a global solution of (1.1) if and only if for all $i \in S$, the following copositivity conditions are satisfied:*

$$Q_{i,j}^{(T)} \quad \textit{is} \quad \Gamma_{i,j}^{(T)}\text{-copositive for all } j \in T\,.$$

Moreover, in the negative case we obtain the following improving feasible direction (cf. Theorem 1): If $z \in \Gamma_{i,j}^{(T)}$ satisfies $z'Q_{i,j}^{(T)}z < 0$ for some $j \in T$, then $v \in \mathbb{R}^n$ with co-ordinates

$$v_s = \begin{cases} -e'z, & \text{if } s = j,\\ 0, & \text{if } s \in T\setminus\{j\},\\ z_s, & \text{if } s \in \mathcal{V}\setminus T,\end{cases}$$

satisfies $v \in \Gamma_i$ and $v'Q_i v < 0$.

Proof. Reconsider the proof of Theorem 12 in [2], which applies to the case of finding a maximum clique where $A = A_{\mathcal{G}} + \frac{1}{2}I$ (with $A_{\mathcal{G}}$ the adjacency matrix of the underlying graph, see Section 5 below) and the current best feasible point (corresponding to the current maximal clique S with k elements) is x with $x_i = 1/k$ if $i \in S$ while $x_i = 0$ otherwise. It is easy to see that the matrix D_i defining Γ_i remains almost the same, the only difference concerning the block R which reads in general even simpler: $R = [I\,|\,-\frac{1}{x_i}\widehat{x}]$, where x is also partitioned as $x' = [o'\,|\,(\widehat{x})']$ according to T and $\mathcal{V}\setminus T \supset S$. Hence also E_I, E_J and F_I remain* as they are in [2] while $F_J' = [-e\,|\,o\,|\,R']$. Hence the definitions used in [1, Theorem 6] yield now,

* As a subscript, *I* always refers to an index set as in [1], [2].

similar to case (+) in [2], but with an additional straightforward rearrangement argument,

$$\begin{aligned} Q_I^{\square} &= C_i - (E_I^{-1}F_I)'B_i - B_i'E_I^{-1}F_I + (E_I^{-1}F_I)A_iE_I^{-1}F_I \\ &= C_i - [e\,|\,O]P_jB_i - B_i'P_j\begin{bmatrix} e' \\ O \end{bmatrix} - x_i[e\,|\,O]P_jA_TP_j\begin{bmatrix} e' \\ O \end{bmatrix} = Q_{i,j}^{(T)} \end{aligned}$$

as specified in (2.4), where P_j is a square permutation matrix which interchanges the first with the j-th row if premultiplied. Note that we use S and T here instead of σ and τ in [2], and drop superscripts for A_i, B_i, and C_i. Similarly, the condition $H_Iz \geq o$ with $H_I' = [o\,| - e'\,|\,R']$ is equivalent to $z \in \Gamma_i^{(T)}$ as defined in (2.5). Next we deal with the additional requirement that $G_Iz \geq o$ with

$$\begin{aligned} G_I &= (E_I')^{-1}[B_i - A_iE_I^{-1}F_I] \\ &= \begin{bmatrix} \pm 1 & | & o' \\ -e & | & I \end{bmatrix} P_jB_i + x_i \begin{bmatrix} \pm 1 & | & o' \\ -e & | & I \end{bmatrix} P_jA_TP_j \begin{bmatrix} e' \\ O \end{bmatrix} \\ &= \begin{bmatrix} \pm r_j' \\ B_{i,\setminus j} - er_j' \end{bmatrix} + x_i \begin{bmatrix} \pm a_{jj}e' \\ ae' \end{bmatrix}, \end{aligned}$$

where $B_{i,\setminus j}$ consists of the rows r_s', $s \in T \setminus \{j\}$ while $a = [\gamma_{sj}]_{s\in T\setminus\{j\}}$, cf. (2.5) and (2.6). Then obviously $G_Iz \geq o$ if and only if $\pm[r_j'z + (x_ia_{jj})e'z] \geq 0$ and

$$r_s'z \geq r_j'z + (x_i\gamma_{s,j})e'z \geq 0 \quad \text{for all } s \in T \setminus \{j\}\,.$$

Therefore we arrive at the definition of the cones $\Gamma_{i,j}^{(T)}$ as given in (2.6) above, since one can merge both cases (+) and (–) into one copositivity condition as in [2], namely that $Q_{i,j}^{(T)}$ be $\Gamma_{i,j}^{(T)}$-copositive. Similarly, case (0) can be dealt with exactly as in [2]. However, to my regret I have to admit that this case can be incorporated already into the others, making the requirement for $m = 0$ in [2, Theorem 12] – while correct – superfluous. Indeed, the subcone corresponding to this case is again contained in $\{z \in \Gamma_i^{(T)} : e'z = 0\}$ whereas $z'C_iz = z'Q_{i,j}^{(T)}z$ if $e'z = 0$, whence Lemma 2 entails that we have in effect only m copositivity conditions (for fixed i and T) instead of $m+1$ as stated in [2]. This also applies to the general case treated here. Now [1, Theorem 6] yields the result together with Theorem 1 above. Indeed, negative semidefiniteness of $A_i = -x_iA_T$ is required there only to guarantee that the parametrized subproblem (5) in [1] is concave, so that a solution is attained at a vertex of the feasible set. Now for $E = [e\,| - e\,|\,I\,|O]'$ and $F = [e\,| - e\,|\,O\,|R']'$, the feasible set $\{z \in \mathbb{R}^{n-m} : Ez \geq -Fw\} \subset e^{\perp}$, so that concavity of $z'A_iz = (-x_i)z'A_Tz$ is guaranteed also if A_T is merely $e^{\perp}$-copositive, which evidently is equivalent to positive semidefiniteness of PA_TP, where $P = I - \frac{1}{m}ee'$ is the orthoprojector onto $e^{\perp}$. □

Note that Theorem 3 yields in total km subproblems of determining whether or not a given symmetric matrix of order $n-m$ is copositive with respect to a polyhedral cone specified by n linear inequalities. While this characterization of global optimality may in itself be much more simpler to check than the previous one (Theorem 1) due to the fact that the effort of checking copositivity increases exponentially with the number of variables involved, we shall proceed to further simplification by showing that these copositivity conditions are equivalent to copositivity of all $Q_{i,j}^{(T)}$ *on the whole of* $\Gamma_i^{(T)}$ rather than on the subcones $\Gamma_{i,j}^{(T)}$. Now the former cone incorporates only $n-m$ restrictions which can explicitly reformulated into positivity requirements on the co-ordinates, so that the conditions in Theorem 3 can be rephrased into those with respect to the positive orthant $\mathbb{R}_+^{n-m}$. Since copositivity is a homogeneous property, it suffices to check whether it holds on Δ^{n-m} instead on the whole of $\mathbb{R}_+^{n-m}$. This will then yield the auxiliary problems of the same type, but with smaller dimension, as claimed above.

THEOREM 4. *If* $z \in \Gamma_i^{(T)}$ *satisfies* $z'Q_{i,s}^{(T)}z < 0$ *for some* $s \in T$, *and if* $j \in T$ *is chosen as in (2.7), then* z *yields an improving feasible direction* v *as specified in Theorem 3.*

Proof. From Lemma 2 we deduce $z \in \Gamma_{i,j}^{(T)}$ and $z'Q_{i,j}^{(T)}z \leq z'Q_{i,s}^{(T)}z < 0$ yields the result. □

The next section is devoted to rephrasing the problem of finding a point z satisfying the above assumptions into a problem of the same type as (1.1). Then we search for a (local) solution and repeat this, cycling over all $s \in T$, and all $i \in S$. The price we have to pay for applying just local optimization procedures to these considerably simpler problems is that in case of failure (i.e. if no improving feasible point is returned), we still have no certificate of global optimality of the current best feasible point x. In order to achieve this, either one would have to use one of the several procedures for detecting copositivity which return a direction violating this condition in case it is not true [3], [14], [15], [16], or to iterate the above procedure recursively, as indicated in Section 4 below.

3. From Block Pivoting to Auxiliary Problems Yielding Global Improvement

Let us start with a useful observation on inverses of rank-two updates of the identity matrix:

LEMMA 5. *Let* f, g, h *be vectors in* $\mathbb{R}^{n-m}$ *satisfying* $f'f = f'g = f'h = 1$ *and* $g'h = \alpha \neq 0$. *Then*

$$(I - fg' - hf')^{-1} = I - ff' - \frac{1}{\alpha}hg'\,.$$

Proof. Straightforward by calculation, see also, e.g. [9, Corollary 5, p.39] with $C = I$; $X = [f \,|\, h]$; $D = \left[\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right]$; and $Y = [g \,|\, f]'$. □

Now we specialize the above result to transform $\Gamma_i^{(T)}$ into $\mathbb{R}_+^{n-m}$: to this end, put $f = \hat{e}_i$, the truncated i-th standard basis vector (we retain f to avoid too many indices); $g = e$, the vector of unit entries; and $h = \frac{1}{x_i}\hat{x}$. Then $\frac{1}{\alpha} = x_i$, and $U = I - fg' - hf'$ yields $u = Uz \geq o$ if and only if $z \in \Gamma_i^{(T)}$ according to Definition (2.5). Furthermore, we have

$$z' Q_{i,s}^{(T)} z = x_i u' R_{i,s} u \quad \text{where} \quad R_{i,s} = \frac{1}{x_i}(U^{-1})' Q_{i,s}^{(T)} U^{-1}$$
$$\text{with} \quad U^{-1} = I - ff' - \hat{x}e' \,.$$

We now have to calculate $R_{i,s}$. To this end, we first do the leftmost product:

$$(U^{-1})' Q_{i,s}^{(T)} = (U^{-1})'[C_i - e{r_s}' - r_s e' - (x_i a_{ss}) ee'] \,. \tag{3.1}$$

First observe that straightforward calculation with partitioned matrices yields more information about the blocks A_i, B_i and C_i of Q_i. Indeed, denoting the parts of A by

$$A = \begin{bmatrix} A_T & V \\ V' & A_C \end{bmatrix} \quad \text{so that} \quad \begin{bmatrix} A_i & B_i \\ B_i' & C_i \end{bmatrix} = Q_i = e_i(Ax)' + (Ax)(e_i)' - x_i A \,,$$

then, as already used above, $A_i = -x_i A_T$ and obviously $x'Ax = (\hat{x})' A_C \hat{x}$. Furthermore, $B_i = V\hat{x}f' - x_i V$ and thus $B_i \hat{x} = V\hat{x}(x_i) - x_i V\hat{x} = o$ while $r_s' = (V\hat{x})_s f' - x_i v_s'$ where $v_s = [a_{sj}]_{j \notin T}$. This entails

$$r_s'\hat{x} = (\hat{e}_s)' B_i \hat{x} = 0 \quad \text{while} \quad r_s' f = (V\hat{x})_s - x_i a_{si} \quad \text{for all } s \in T \,. \tag{3.2}$$

Finally, we obtain the explicit form of C_i:

$$C_i = f(A_C\hat{x})' + (A_C\hat{x})f' - x_i A_C \,. \tag{3.3}$$

Now using Lemma 5 we derive $(U^{-1})'e = [I - ff' - e(\hat{x})']e = (1-1)e - f = -f$ and similarly

$$\begin{aligned} (U^{-1})' r_s &= [I - ff' - e(\hat{x})'] r_s = r_s - (r_s' f) f - 0e \\ &= (V\hat{x})_s f - x_i v_s - [(V\hat{x})_s - x_i a_{si}] f = x_i [a_{si} f - v_s] \,. \end{aligned}$$

Hence we obtain from (3.1), collecting terms,

$$(U^{-1})' Q_{i,s}^{(T)} = (U^{-1})' C_i + f r_s' - x_i [(a_{si} - a_{ss}) f - v_s] e' \,.$$

Proceeding similarly with the right factor, we now use $r_s' U^{-1} = x_i [a_{si} f' - v_s']$ and $e' U^{-1} = -f'$ to obtain

$$\begin{aligned} x_i R_{i,s} &= (U^{-1})' C_i U^{-1} + x_i f [a_{si} f' - v_s'] + x_i [(a_{si} - a_{ss}) f - v_s] f' \\ &= (U^{-1})' C_i U^{-1} + x_i [(2a_{si} - a_{ss}) ff' - v_s f' - f v_s'] \,. \end{aligned} \tag{3.4}$$

LEMMA 6. *For* $R_{i,s} = \frac{1}{x_i}(U^{-1})' Q_{i,s}^{(T)} U^{-1}$ *we have*

$$\begin{aligned} R_{i,s} \;=\; & (x'Ax)ee' - (I - ff')A_C(I - ff') \\ & + [2a_{si} - a_{ss}]ff' - v_s f' - f v_s' \end{aligned} \tag{3.5}$$

with $v_s = [a_{sj}]_{j \notin T}$ *and* $f = \widehat{e_i}$.

Proof. It remains to determine $(U^{-1})' C_i U^{-1}$. From (3.3) we get $C_i f = (A_C \widehat{x})_i f + (A_C \widehat{x}) - x_i A_C f$ as well as $C_i \widehat{x} = (\widehat{x})' A_C \widehat{x} f + (A_C \widehat{x}) x_i - x_i (A_C \widehat{x}) = (x'Ax)f$. Hence

$$\begin{aligned} C_i U^{-1} \;&=\; C_i - C_i f f' - C_i \widehat{x} e' \\ &=\; C_i - [(A_C \widehat{x})_i I - x_i A_C] f f' - (A_C \widehat{x}) f' - (x'Ax) f e' \,, \end{aligned} \tag{3.6}$$

and therefore, rearranging terms

$$\begin{aligned} (U^{-1})' C_i U^{-1} \;=\; & (U^{-1})' C_i - (U^{-1})' f [(A_C \widehat{x})_i f' + (x'Ax) e'] \\ & + (U^{-1})' A_C [x_i f f' - (\widehat{x}) f'] \,. \end{aligned}$$

Now $(U^{-1})' f = -x_i e$, and also $(U^{-1})' A_C = [I - ff'] A_C - e (A_C \widehat{x})'$ according to Lemma 5, while transposing (3.6) gives

$$(U^{-1})' C_i = C_i - f f' [(A_C \widehat{x})_i I - x_i A_C] - f (A_C \widehat{x})' - (x'Ax) e f' \,.$$

Plugging these three terms into (3.7) we finally obtain, using again (3.3),

$$\begin{aligned} (U^{-1})' C_i U^{-1} \;=\; & C_i - f f' [(A_C \widehat{x})_i I - x_i A_C] - f (A_C \widehat{x})' - (x'Ax) e f' \\ & + x_i e [(A_C \widehat{x})_i f' + (x'Ax) e'] \\ & + [I - ff'] A_C [x_i f f' - \widehat{x} f'] \\ & - e (A_C \widehat{x})' [x_i f f' - \widehat{x} f'] \\ =\; & f (A_C \widehat{x})' + (A_C \widehat{x}) f' - x_i A_C \\ & - (A_C \widehat{x})_i f f' + x_i f f' A_C - f (A_C \widehat{x})' - (x'Ax) e f' \\ & + x_i (A_C \widehat{x})_i e f' + x_i (x'Ax) e e' \\ & + x_i [I - ff'] A_C f f' - (A_C \widehat{x}) f' + (A_C \widehat{x})_i f f' \\ & - x_i (A_C \widehat{x})_i e f' + (x'Ax) e f' \\ =\; & x_i [(x'Ax) e e' - (I - ff') A_C (I - ff')] \end{aligned}$$

which together with (3.4) yields the result. □

Note that whenever $u = Uz$ satisfies $u_i = -e'z = 0$, we get from (3.5)

$$z' Q_{i,s}^{(T)} z = x_i \, u' R_{i,s} u = x_i [x'Ax - u' A_C u] \,,$$

so that in this case the improvement result of Theorem 3 is nothing else than the requirement that there is an improving feasible point $\tilde{x}$ with $\tilde{x}_j = 0$ for all $j \in T$, cf. [2, Theorem 7].

For the sake of transparency, we recapitulate our findings in the following

THEOREM 7. *Suppose that $x \in \Delta^n$ is a local solution to (1.1), and denote by $S = \{i \in \mathcal{V} : x_i > 0\}$ the set of its positive co-ordinates with #$S = k$. Pick a disjoint subset $T \subset \mathcal{V} \setminus S$ of size $m \leq n - k$ such that $(I - \frac{1}{m}ee')A_T(I - \frac{1}{m}ee')$ is positive semidefinite. Then x is a global solution of (1.1) if and only if for all $i \in S$, the following km QPs in $\mathbb{R}^{n-m}$ have objective values which do not exceed the current best value $x'Ax$:*

$$u'A^{(T)}_{i\leftarrow s}u \to \max ! \quad \textit{subject to } u \in \Delta^{n-m}\,, \tag{3.7}$$

where $i \in S$; $s \in T$; and $A^{(T)}_{i\leftarrow s}$ is the symmetric matrix of order $n - m$ obtained by deleting all rows and columns belonging to indices in T and replacing the i-th row and column by the s-th row and column. If $u'A^{(T)}_{i\leftarrow s}u > x'Ax$ for some $u \in \Delta^{n-m}$ and $j \in T$ is chosen such that

$$\sum_{p\notin T\cup\{i\}} a_{jp}u_p + \tfrac{1}{2}a_{jj}u_i \geq \sum_{p\notin T\cup\{i\}} a_{qp}u_p + \tfrac{1}{2}a_{qq}u_i \quad \textit{for all } q \in T\,, \tag{3.8}$$

then a strictly improving feasible point $\tilde{x}$ is obtained as follows:

$$\tilde{x}_q = \begin{cases} u_i, & \text{if } q = j,\\ 0, & \text{if } q \in T \cup \{i\} \setminus \{j\},\\ u_q, & \text{if } q \in \mathcal{V} \setminus T. \end{cases}$$

Proof. First note that straightforward calculation shows $u'R_{i,s}u = x'Ax - u'A^{(T)}_{i\leftarrow s}u$ on Δ^{n-m}, and that $\mathbb{R}^{n-m}_+$-copositivity of $R_{i,s}$ is equivalent to the property that $u'R_{i,s}u$ takes only nonnegative values on Δ^{n-m}. In light of the preceding arguments, we have only to derive the improving feasible direction v from Theorems 1 and 3. Now for $u \in \Delta^{n-m}$, we get $z = U^{-1}u = u-(f'u)f-(e'u)\widehat{x} = u-u_if-\widehat{x}$, so that

$$z_q = \begin{cases} -x_i, & \text{if } q = i,\\ u_q - x_q, & \text{if } q \in \mathcal{V} \setminus (T \cup \{i\}), \end{cases}$$

while Theorem 3 entails, by virtue of $-e'z = u_i$,

$$v_q = \begin{cases} u_i, & \text{if } q = j,\\ 0, & \text{if } q \in T \setminus \{j\},\\ u_q - x_q, & \text{if } q \in \mathcal{V} \setminus (T \cup \{i\}),\\ -x_i, & \text{if } q = i. \end{cases}$$

Hence $x_i/v_i = -1$ and therefore, from Theorem 1, $\tilde{x} = x + v$ with coordinates specified as above, provided that $z = Uu \in \Gamma^{(T)}_{i,j}$. But this relation follows from Lemma 2 together with $r'_q = (V\widehat{x})_q f' - x_i v'_q$ where $v_q = [a_{qp}]_{p\notin T}$, entailing

$$\begin{aligned} r'_q z + \tfrac{1}{2}x_i a_{qq} e'z \;&=\; (V\widehat{x})_q z_i - x_i v'_q z + \tfrac{1}{2}x_i a_{qq} e'z \\ &=\; -x_i[(V\widehat{x})_q + (Vz)_q + \tfrac{1}{2}a_{qq}u_i] \end{aligned}$$

$$\begin{aligned} &= -x_i[(V\widehat{x})_q + [V(u - u_i f - \widehat{x})]_q + \tfrac{1}{2}a_{qq}u_i] \\ &= -x_i[(Vu)_q - u_i a_{qi} + \tfrac{1}{2}a_{qq}u_i] \\ &= -x_i[\sum_{p\in T\setminus\{i\}} a_{qp}u_p + \tfrac{1}{2}a_{qq}u_i]\,. \end{aligned}$$

Hence (3.9) guarantees $z \in \Gamma^{(T)}_{i,j}$, and Theorem 4 together with $z'Q^{(T)}_{i,s}z = x_i u' R_{i,s} u < 0$ proves strict improvement: $(\tilde{x})'A\tilde{x} > x'Ax$. □

4. Local Minimizers Yield Good Pivots; The Algorithm

Now given $S \subset \mathcal{V}$ we have to find a subset $T \subseteq \mathcal{V} \setminus S$ such that PA_TP is positive semidefinite where $P = I - \frac{1}{m}ee'$ is the orthoprojector onto $e^\perp$. To this end we follow a counter-greedy approach in that we try to obtain a local *minimizer* of $x'Ax$, which means to consider the auxiliary QP

$$y'\widehat{A}y \to \max! \quad \text{subject to } x \in \Delta^{n-k}\,, \tag{4.1}$$

where $\widehat{A} = \gamma_S ee' - [a_{ij}]_{i,j\in\mathcal{V}\setminus S}$ with $\gamma_S = \max\{a_{sj} : s, j \in \mathcal{V} \setminus S\}$ to ensure that $\widehat{A}$ has non-negative entries. Then apply, e.g., the algorithm described in [5], in order to obtain a local solution to (4.1). The following result guarantees that this counter-greedy approach yields a submatrix A_T having the properties required in Theorem 7 to obtain global improvement (or to prove global optimality of the current solution).

THEOREM 8. *If y is a local solution of (4.1) and $T = \{j \in \mathcal{V} \setminus S : y_j > 0\}$, then PA_TP is positive semidefinite.*

Proof. We use the characterization of local optimality in QPs due to Contesse [7] and Borwein [6] in the formulation of [8, Theorem 1], whence it follows that $-\widehat{A}$ is Γ^*-copositive with $\Gamma^* = \{v \in \Gamma : v'(\widehat{A}y) = 0\}$, where

$$\Gamma = \{v \in e^\perp : v_j = 0 \text{ if } j \in \mathcal{V} \setminus (S \cup T)\}\,.$$

Now from local optimality of y we infer that necessarily $v'(\widehat{A}y) = 0$ for all $v \in \Gamma$, see, e.g. [4, Theorem 2]. Hence $\Gamma^* = \Gamma$. Partitioning w.r.t. T and $\mathcal{V} \setminus (S \cup T)$, we see that every $v \in \Gamma$ can be written as $v = \begin{bmatrix}\widehat{v}\\ o\end{bmatrix}$ with $\widehat{v} = Pa$ for some $a \in \mathbb{R}^m$, and vice versa. Hence we obtain $a'PA_TPa = -v'\widehat{A}v \geq 0$, whence the assertion follows. □

The algorithm is now straightforward. For beauty of exposition, we formulate it in a recursive way, although for practical implementation one would have to restrict the depth of recurrence to prevent combinatorial explosion in hard instances. If local solutions are generated by following paths under the replicator dynamics as in [2], finiteness follows from the results in [5] under simple regularity conditions.

1. Starting from a suitable point $x(0) \in \Delta^n$, find a local solution x of (1.1), put $S = \{i : x_i > 0\}$ and $\gamma = x'Ax$;
2. similarly, find a local solution y of (4.1), put $T = \{i : y_i > 0\}$ and $m = \#T$; then PA_TP is positive semidefinite (Theorem 8);
3. for all $i \in S$ and all $s \in T$, find a local solution $u_{i,s}$ of (4.8), until $u'_{i,s}A^{(T)}_{i\leftarrow s}u_{i,s} > \gamma$. Then form an improving feasible point $\tilde{x}$ for (1.1) as in Theorem 7; replace $x(0)$ with $\tilde{x}$ and go to step 1;
4. if $n > 1$ and all auxiliary problems (4.8) yield objective values not exceeding γ, then replace n with $n - m$; A with $A^{(T)}_{i\leftarrow s}$, and go to step 1; again, cycle over all $i \in S$ and $s \in T$ until γ is exceeded;
5. else Theorem 7 guarantees that the current local solution x is the global one.

5. Application: Search for a Maximum Clique

Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\#\mathcal{V} = n$ nodes. A *clique* S is a subset of the node set $\mathcal{V}$ which corresponds to a complete subgraph of $\mathcal{G}$ (i.e., any pair of nodes in S is an edge in $\mathcal{E}$, the edge set). A clique S is said to be *maximal* if there is no larger clique containing S. A (maximal) clique is said to be a *maximum* clique if it contains most elements among all cliques. The search for such a maximum clique is an NP-hard problem, for a concise survey see, e.g. [11].

Now suppose some algorithm returns a maximal clique S which is not a maximum clique, and denote by S^* a maximum clique. Of course, there must be a node $i \in S \setminus S^*$, so that a naive strategy would be restarting the employed algorithm on the graph with node i removed, i.e. for $\mathcal{G}_i = (\mathcal{V} \setminus \{i\}, \mathcal{E} \setminus \{(i,j),(j,i) : j \in \mathcal{V}\})$, and repeating this, cycling over all $i \in S$. While this procedure has its merits from a practical viewpoint [2], dimensionality of the problem is reduced only by one. The situation is different in the procedure proposed here, which similarly cycles over all $i \in S$, but considers auxiliary problems of even smaller dimension. Hence, the hope to obtain larger cliques with this approach is justified, at least to a larger extent than with the naive strategy.

As shown in [2], the maximum clique problem can be reformulated into (1.1) with $A = \frac{1}{2}I + A_{\mathcal{G}}$ where $A_{\mathcal{G}} = [a_{ij}]_{i,j\in\mathcal{V}}$ is the $n \times n$ adjacency matrix of $\mathcal{G}$, i.e. $a_{ij} = 1_{\mathcal{E}}(i,j)$ for all (i,j). Hence $\widehat{A}$ from (4.1) coincides with $\frac{1}{2}I + A_{\overline{\mathcal{G}}|\mathcal{V}\setminus S}$, where $\overline{\mathcal{G}}|\mathcal{V} \setminus S$ denotes the complementary graph $\overline{\mathcal{G}}$ restricted to the node set $\mathcal{V} \setminus S$. Therefore solving (4.1) means searching for a maximal clique T of $\overline{\mathcal{G}}|\mathcal{V} \setminus S$, i.e. a maximal independent set $T \subseteq \mathcal{V} \setminus S$ of vertices in $\mathcal{G}$. This case is treated to the extent of Theorem 3 in [2, Theorem 12], which unfortunately contains a misprint: there $\frac{1}{2m}$ should be replaced with $\frac{1}{2k}$ in the definition of $Q^{(\tau)}_{i,j}$ and $\Gamma^{(\tau)}_{i,j}$. However, also Theorem 7 can be simplified in this case. For convenience, we formulate the result in terms of maximal cliques and maximal independent sets (of course, determination of these can again be accomplished by means of the replicator dynamics as in [2]), and note that a purely combinatorial proof of this – at least the necessity part with (5.1) – is seemingly not quite immediate.

THEOREM 9. *Suppose that $S \subset \mathcal{V}$ is a maximal clique of size k in a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with adjacency matrix $A_\mathcal{G} = [a_{ij} = 1_\mathcal{E}(i,j)]_{i,j\in\mathcal{V}}$. Pick a disjoint independent set $T \subset \mathcal{V} \setminus S$ of size $m \leq n-k$. Denote by $\mathcal{G}^{(T)}_{i\leftarrow s}$ the graph of order $n-m$ obtained from $\mathcal{G}$ with all nodes in T removed, and the roles of nodes i and s interchanged.*
Then S is a maximum clique of $\mathcal{G}$ if and only if for all $i \in S$, all $s \in T$ the graphs $\mathcal{G}^{(T)}_{i\leftarrow s}$ have a maximum clique with not more than k elements. If $U \subset \mathcal{V} \setminus T$ is a maximal clique of $\mathcal{G}^{(T)}_{i\leftarrow s}$ with size larger than k, and $j \in T$ is chosen such that

$$d_{U\setminus\{i\}}(j) = \sum_{p\in U\setminus\{i\}} a_{jp} \geq \sum_{p\in U\setminus\{i\}} a_{qp} = d_{U\setminus\{i\}}(q) \quad \textit{for all } q \in T\,, \tag{5.1}$$

then either $(U \cup \{j\}) \setminus \{i\}$ or U is a larger clique than S, depending on whether $i \in U$ or not.

Proof. Theorem 9 of [2] entails that every local solution u of (4.1) here has the form $u = b_U$ for some $U \subseteq \mathcal{V} \setminus T$, i.e.

$$u_q = \begin{cases} \frac{1}{\#U}, & \text{if } q \in U, \\ 0, & \text{otherwise.} \end{cases}$$

The remainder is an easy consequence of Theorem 7, since here $a_{qq} = \frac{1}{2}$ for all $q \in T$. □

To assess the effectiveness of the proposed procedure, extensive simulations are necessary which still are work in progress. For a more detailed report on the project we refer to [5], which also will contain the data presented here. In this preliminary phase of the study 22 selected DIMACS benchmark graphs were investigated. All these instances already have been considered in [4]. The local optimization part used the discrete time version of the replicator equation with $A = A_\mathcal{G} + \frac{1}{2}I$ which frequently is called the Comtet approach, for details see [2].

The results of the simulations are summarized in Table I containing, for each problem instance, indicated in the column labeled “Graph” by the file name with suppressed suffix `.clq(.b)`; the order n (number of nodes), density (“Dens.”), i.e. the ratio of the number of edges by the maximum number $\binom{n}{2}$; the actual size of the maximum cliques (column labeled “Max Cli.”) with the exception of the last instance where only a lower bound is known; the size of the clique obtained by local search; first improvement; and final improvement, the latter two as a result of recursive application of the algorithm as described in Section 4.

To illustrate runtime behaviour, the last two columns contain the ratios of time used to obtain the first improved result relative to that used to get the local solution, and overall time consumed relative that used for the first improvement.

Table 1. Results on DIMACS benchmark graphs. 46*: best known value

Graph	n	Dens.	Max Cli.	local result	improvements first	improvements final	rel.time first	rel.time final
mann_a9	45	0.927	16	12	–	16	–	782.6
keller4	171	0.649	11	7	8	9	08.36	439.5
san200_0.7_1	200	0.700	30	15	–	17	–	131.2
san200_0.7_2	200	0.700	18	12	–	14	–	21.79
san200_0.9_1	200	0.900	70	45	46	47	99.26	22.55
san200_0.9_2	200	0.900	60	36	38	40	03.23	10.15
san200_0.9_3	200	0.900	44	32	34	35	260.0	61.71
san400_0.5_1	400	0.500	13	7	–	12	–	03.78
san400_0.7_1	400	0.700	40	20	–	21	–	06.70
san400_0.7_2	400	0.700	30	15	16	17	31.84	169.4
san400_0.7_3	400	0.700	22	12	14	15	66.68	33.66
san400_0.9_1	400	0.900	100	52	54	56	67.03	42.35
sanr200_0.7	200	0.697	18	14	17	18	34.86	340.2
sanr400_0.5	400	0.501	13	11	–	12	–	807.5
brock200_1	200	0.745	21	17	18	20	04.50	3,927
brock200_3	200	0.605	15	9	12	13	13.67	1,964
brock200_4	200	0.658	17	12	13	16	03.66	23,564
p_hat300-1	300	0.244	8	6	7	8	11.67	142.9
p_hat300-2	300	0.489	25	22	23	25	08.41	13.48
p_hat300-3	300	0.744	36	32	33	34	01.72	03.88
p_hat500-1	500	0.253	9	8	–	9	–	03.54
p_hat1000-2	1000	0.490	46*	42	43	44	03.92	23.33

The code was written in the C programming language and run on a PC (486/66 DX2) under UNIX-Solaris System V (no attempt was made to optimize the code). Due to time constraints, recursion depth was bounded by two, so that in some cases a single improvement has been obtained.

As can be seen, the results obtained are fairly encouraging. The quality of the cliques (final improvement/Max Cli.) range from 52% to 100%, with 13 cases out of 22 exceeding 80% and a vast majority (18) exceeding 66 %. Compared to other continuous approaches like Pelillo's relaxation labeling network technique [13] or the continuous-based heuristic by Gibbons et al. [10], the procedure proposed here is beaten only 4 times (by at most 2 nodes) while dominating these in other instances by up to 5 nodes, as a comparison with the figures in [4] shows. This is particularly remarkable in view of the different hardware situation, which is also the reason why absolute runtime data are not very informative and therefore omitted here. A detailed simulation also over random graphs is currently carried out [5] and will shed more light on the – as we expect – advantageous average performance of the presented procedure.

Acknowledgements

The author benefited from valuable suggestions by two anonymous referees, and is indebted to V. Stix for his assistance with the benchmark results.

References

1. Bomze, I.M. (1996), Block pivoting for detecting copositivity. *Linear Algebra Appl.* **248**, 161–184.
2. Bomze, I.H. (1997), Evolution towards the maximum clique, *J. Global Optim.* **10**, 143–164.
3. Bomze, I.H. and Danninger, G. (1993), A global optimization algorithm for concave quadratic programming, *SIAM J. Optim.* **3**, 826–842.
4. Bomze, I.H., Pelillo, M., and Giacomini, R. (1997), Evolutionary approach to the maximum clique problem: empirical evidence on a larger scale. In: I.M. Bomze, T. Csendes, R. Horst, and P.M. Pardalos (eds.), *Developments in Global Optimization*, pp. 95-108. Kluwer, Dordrecht.
5. Bomze, I.H. and Stix, V. (1997), Genetical engineering via negative fitness: evolutionary dynamics for global optimization. I.S.O.C. Report TR 97–02, Univ. of Vienna.
6. Borwein, J.M. (1982), Necessary and sufficient conditions for quadratic minimality, *Numer. Funct. Anal. Optimiz.* **5**, 127–140.
7. Contesse, L.B. (1980), Une caractérisation complète des minima locaux en programmation quadratique, *Numer. Math.* **34**, 315–332.
8. Danninger, G. and Bomze, I.M. (1993), Using copositivity for global optimality criteria in concave quadratic programming problems, *Math. Programming* **62**, 575–580.
9. Dhrymes, Ph.J. (1984), *Mathematics for econometrics*, 2nd ed., Springer, New York.
10. Gibbons, L.E., Hearn, D.W., and Pardalos, P.M. (1996). A continuous based heuristic for the maximum clique problem. In: D.S. Johnson and M.A. Trick (eds.), *Clique, Graph Coloring, and Satisfiability: Second DIMACS Implementation Challenge*, pp. 103–124. DIMACS Series Vol. 26, American Mathematical Society.
11. Horst, R., Pardalos, P.M., and Thoai, N.V. (1995), *Introduction to Global Optimization*, Kluwer, Dordrecht.
12. Murty, K.G., and Kabadi, S.N. (1987), Some NP-complete problems in quadratic and linear programming, *Math. Programming* **39**, 117–129.
13. Pelillo, M. (1995), Relaxation labeling networks for the maximum clique problem. *J. Artif. Neural Networks* **2**, 313–327.
14. Väliaho, H. (1988), Testing the definiteness of matrices on polyhedral cones, *Linear Algebra Appl.* **101**, 135–165.
15. Väliaho, H. (1989), Quadratic-programming criteria for copositive matrices, *Linear Algebra Appl.* **119**, 163–182.
16. Väliaho, H. (1992), Determining subspaces on which a matrix is nonnegative definite. *Linear Algebra Appl.* **168**, 71–93.