Annealed Replication:
A New Heuristic for the Maximum Clique Problem*

Immanuel M. Bomze

Institut für Statistik, Operations Research und Computerverfahren

Universität Wien

Universitätsstraße 5, A-1010 Wien, Austria

bomze@smc.univie.ac.at

Marco Budinich

Dipartimento di Fisica

Università di Trieste

Via Valerio 2, I-34127 Trieste, Italy

mbh@trieste.infn.it

Marcello Pelillo **Claudio Rossi**

Dipartimento di Matematica Applicata e Informatica

Università "Ca' Foscari" di Venezia

Via Torino 155, I-30173 Venezia Mestre, Italy

{pelillo,rossi}@dsi.unive.it

April 23, 1998

*Submitted to *Discrete Applied Mathematics*, Special issue on "Discrete vs analog computation"

Abstract

In this paper, a new heuristic for approximating the maximum clique problem is proposed, based on a detailed analysis of a class of continuous optimization models which yield a complete solution to this NP-hard combinatorial problem. The idea is to alter a regularization parameter iteratively in such a way that an iterative procedure with the updated parameter value would avoid unwanted, inefficient local solutions, i.e., maximal cliques which contain less than the maximum possible number of vertices. The local search procedure is performed with the help of the replicator dynamics, and the regularization parameter is chosen deliberately as to render dynamical instability of the (formerly) stable solutions which we want to discard in order to get an improvement. In this respect, the proposed procedure differs from usual simulated annealing approaches which mostly use a "black-box" cooling schedule. To demonstrate the validity of this approach, we report on the performance applied to selected DIMACS benchmark graphs.

1 Introduction

The maximum clique problem (MCP) is a well-known problem in combinatorial optimization which finds important applications in many different domains [41]. Unfortunately, the MCP is known to be NP-complete, and therefore exact algorithms are guaranteed to return a solution only in a time which increases exponentially with the number of vertices in the graph. This makes them inapplicable even to moderately large problem instances. Moreover, a series of recent theoretical results show that the MCP is in fact difficult to solve even in terms of approximation. Strong evidence of this fact came in 1991, when Feige *et al.* [17] (see also [18]) proved that if there is a polynomial-time algorithm that approximates the MCP within a factor of $2^{\log^{1-\epsilon} n}$, then any NP problem can be solved in "quasi-polynomial" time (i.e., in $2^{\log^{O(1)} n}$ time). The result was further refined by Arora *et al.* [1, 2] one year later. Specifically, they proved that there exists an $\epsilon > 0$ such that no polynomial-time algorithm can approximate the size of the maximum clique within a factor of n^ϵ, unless $P = NP$. More recent developments along these lines can be found in [3, 4, 24]. In light of these negative results, much effort has recently been directed towards devising efficient heuristics for the MCP, for which no formal guarantee of performance may be provided, but are anyway of interest in practical applications. We refer to [27] for a recent collection of promising heuristics for the MCP.

We have recently investigated the effectiveness of an approach for approximating the MCP, centered around a continuous formulation due to Motzkin and Straus [37] and Comtet [15], which exploits the dynamical properties of the so-called *replicator equations*, a class of dynamical systems developed and studied in various branches of mathematical biology. One problem associated with these models, however, is their inability to escape inefficient local solutions. In this paper, we introduce a class of parametrized quadratic programs, which includes both the Motzkin-Straus and the Comtet programs as special cases, and investigate the properties of its solutions as a function of its parameter. A detailed analysis of these properties suggests a new algorithm for approximating the MCP which is based on the idea of properly varying the parameter during the replicator optimization process, in much the same spirit as simulated annealing procedures. A related, but different, idea has recently been proposed by Gee and Prager in the neural network domain [21]. Experimental results conducted on various DIMACS benchmark graphs demonstrate the validity of the proposed approach.

The outline of the paper is as follows. In Section 2, we describe the Motzkin-Straus theorem and its parametrization, and present the replicator dynamical systems. These dynamics are used to obtain locally optimal solutions to the MCP. Section 3 is devoted to a few results that enable us to establish bounds on a regularization parameter α which governs stability under the replicator dynamics. For illustration, we investigate in Section 4 two small, but prototypical examples in detail. Rather than with the usual perturbation theory approach, we specify explicit ranges within which qualitative features of the dynamics are invariant, and also obtain quantitative sensitivity results for the related optimization problems. The established theoretical properties will lead us to develop in Section 5 an algorithm for properly updating the parameter α with the objective of avoiding poor local solutions. In Section 6 the results of our experiments are presented, and Section 7 concludes the paper.

2 Evolution towards the maximum clique

2.1 A parametrized continuous formulation of the MCP

Let $G = (V, E)$ be an undirected graph, where $V = \{1, \cdots, n\}$ is the set of vertices, and $E \subseteq V \times V$ is the set of edges. A *clique* of G is a subset of V in which every pair of vertices is connected by an edge. A clique is called *maximal* if no strict superset of C is a clique, i.e., no vertex external to C is connected with more than $|C| - 1$ vertices of C. Here, and in the sequel, $|S|$ denotes the cardinality of a set S. A maximal clique C is called *strictly maximal* if no vertex i external to C has the property that the enlarged set $S \cup \{i\}$ contains a clique of the same size as C. A *maximum* clique is a clique having largest cardinality (note that a maximal clique is not necessarily a maximum one). Hence, the MCP consists of finding a clique of maximum size in a graph G [41]. In the following, given a set S of vertices in G, we will denote by x^S its characteristic vector, defined as $x_i^S = 1/|S|$ if $i \in S$ and $x_i^S = 0$ otherwise.

Given a graph G, consider the following quadratic program introduced by Comtet [15] in a textbook exercise:

$$\begin{array}{ll} \text{maximize} & f(x) = x'(A_G + \frac{1}{2}I)x \\ \text{subject to} & x \in S^n \end{array} \tag{1}$$

where $A_G = (a_{ij})$ is the adjacency matrix of G (i.e., $a_{ij} = 1$ if $(i,j) \in E$, and $a_{ij} = 0$ if $(i,j) \notin E$), S^n is the standard simplex of $\mathbb{R}^n$, that is

$$S^n = \left\{ \mathbf{x} \in \mathbb{R}^n \ : \ x_i \geq 0 \ \text{ for all } i = 1 \dots n, \text{ and } \sum_{i=1}^{n} x_i = 1 \right\} .$$

and I is the $n \times n$ identity matrix. This turns out to be a variant of the so-called Motzkin-Straus program [37], which is obtained from (1) by simply dropping the $\frac{1}{2}I$ term.

The following theorem, proved in [11], is a sharpening of the Motzkin-Straus theorem [37]. It establishes a one-to-one correspondence between local/global maxima of (1) and maximal/maximum cliques of G, respectively.

Theorem 1 *Let G be a graph and consider problem (1). Then the following assertions are equivalent:*

(a) $x = x^C$, where C is a maximal clique of size $k = |C|$;

(b) x is a strict local solution to (1);

(c) x is a local solution to (1).

If one of the above conditions (and therefore all) is met, the objective is $x'(A_G + \frac{1}{2}I)x = 1 - \frac{1}{2k}$. Hence C is a maximum clique of G if and only if x^C is a global solution to (1).

The previous result has an intriguing computational significance. It suggests a fundamentally new way of solving the maximum clique problem, by allowing us to shift from the discrete to the continuous domain. A pointed out by Pardalos [39], the advantages of such a reformulation are manifold. It not only allows us to exploit the full arsenal of continuous optimization techniques, thereby leading to the development of new efficient algorithms, but may also reveal unexpected theoretical properties. Additionally, continuous optimization methods are sometimes described in terms of sets of differential equations, and are therefore potentially implementable in analog circuitry. The Motzkin-Straus and related theorems have served as the basis of many clique-finding procedures [40, 42, 22, 12], and have also been used to determine theoretical bounds on the maximum clique size [14].

In contrast to the original Motzkin-Straus formulation, however, the Comtet program has a further merit. In fact, as observed by Pardalos and Phillips [40] and later formalized by Pelillo and Jagota [45], the Motzkin-Straus program, in its original formulation, is plagued by the presence of "spurious" solutions, i.e., solutions which are not in the form of characteristic vectors. In principle, spurious solutions represent a problem since do not allow us to extract the vertices comprising the clique, but provide information only on its *size*. Therefore, in order to determine the clique vertices, one has to make recourse to iterative or recursive procedure, as those described in [38, 40]. Of course, this dramatically slows down the process. The significance of the previous result is that a local (or global) maximum of (1) can only be attained at a characteristic vector $x^* = x^C$ of the form $x_i^* = 1/k$ if $i \in C$ and $x_i^* = 0$ if $i \notin C$, for some subset C of vertices. This solves the spurious solution problem in a straightforward and definitive manner.

In this paper, we consider the following program:

$$\begin{array}{ll} \text{maximize} & f_\alpha(x) = x'(A_G + \alpha I)x \\ \text{subject to} & x \in S^n \end{array} \tag{2}$$

which includes both the Motzkin-Straus ($\alpha = 0$) and the Comtet ($\alpha = \frac{1}{2}$) programs as special cases. We investigate the properties of its solutions as a function of the parameter α. Specifically, we show that when $\alpha \in]0, 1[$ all the properties of the Comtet program (1) hold true. For negative α, instead, the landscape of $f_\alpha(x)$ changes and "flat regions" can merge in an extremum while other extrema can disappear, depending on the values of the parameter α. A detailed analysis of these effects will suggest a new algorithm for approximating the MCP which is based on the idea of varying the parameter α during an evolutionary optimization process, in such a way as to avoid obtaining characteristic vectors of small cliques.

We point out that the proposed parametrization of the Motzkin-Straus program is completely different, both in content and motivations, from that recently introduced by Gibbons *et al.* [23]. Their idea was to substitute the sign constraints $x \geq 0$ of the Motzkin-Straus program with one of the form $x'x = 1/s$, s being a parameter in the interval $[1, n]$, in an attempt to avoid spurious solutions. With this program it may happen that the solutions have to be projected onto the positive orthant, in order to maintain feasibility.

2.2 Replicator equations and their application to the MCP

Let M be a non-negative real-valued $n \times n$ matrix, and consider the following dynamical system:

$$\dot{x}_i(t) = x_i(t)[(Mx(t))_i - x(t)'Mx(t)] \,, \quad i = 1 \dots n \tag{3}$$

where a dot signifies derivative w.r.t. time t, and its discrete-time counterpart

$$x_i(t+1) = x_i(t)\frac{(Mx(t))_i}{x(t)'Mx(t)} \,, \quad i = 1 \dots n \,. \tag{4}$$

It is readily seen that the simplex S^n is invariant under these dynamics, which means that every trajectory starting in S^n will remain in S^n for all future times. Moreover, it turns out that their *stationary points*, i.e. the points satisfying $\dot{x}_i(t) = 0$ for (3) or $x_i(t+1) = x_i(t)$ for (4), coincide and are the solutions of the equations

$$x_i[(Mx)_i - x'Mx] = 0 \,, \quad i = 1 \dots n \,. \tag{5}$$

A stationary point x is said to be *asymptotically stable* if every solution to (3) or (4) which starts close enough to x, will converge to x as $t \to \infty$.

Both (3) and (4) are called *replicator equations* in theoretical biology, since they are used to model evolution over time of relative frequencies $x_i(t)$ of interacting, self-replicating entities. Equation (3) has been introduced in evolutionary game theory by Taylor and Jonker [46] to model evolution of behavior in intraspecific conflicts under random pairwise mating in a large, ideally infinite population. It formalizes the idea that the growth rates $\dot{x}_i/x_i$ of relative frequency x_i of the i-th behavior pattern ($i = 1 \dots n$) is equal to the (dis)advantage $(Mx)_i - x'Mx = \sum_j m_{ij}x_j - \sum_{j,k} m_{kj}x_jx_k$, measured by incremental fitness relative to the average performance within the population in state $x = (x_1, \cdots, x_n)'$. Here m_{ij} denotes incremental individual fitness attributed to an i–individual when encountering a j-individual, and $M = (m_{ij})$ is the resulting fitness matrix. The behavior patterns $i \in \{1, \cdots, n\}$ are often called "pure strategies" and the interaction matrix M is also termed "payoff matrix." Similar arguments provide a rationale for the discrete-time version (4). Surprisingly, these dynamical equations can also be regarded as a very special case of a general class of dynamical systems introduced by Baum and Eagon [5] and studied by Baum and Sell [6] in the context of Markov chain theory. This kind of processes have proven to be useful in the speech recognition [33] and computer vision [43] domains. The dynamics (3) and (4) also arise in population genetics under the name *selection equations* in a model assuming separate (non-overlapping) generations, large population size, random union of gametes, and a selection acting only upon one chromosomal locus through different viabilities (i.e., survival probabilities), given by the the fitness matrix M of the genotypes, i.e., pairs of genes drawn from a set $\{1, \cdots, n\}$ of alleles for a single chromosomal locus. Here x_i is the gene frequency of the i-th allele. The matrix M is in this context always symmetric, since permuted gene pairs belong to the same genotype. The models (3) and (4) as selection equations go way back to Fisher [19] and Kimura [31].

From an optimization point of view, the difference between symmetric and non-symmetric matrices M is crucial. Indeed, in the symmetric case the quadratic form $x(t)'Mx(t)$ is increasing along trajectories of the replicator dynamics; this is the Fundamental Theorem of Natural Selection, see, e.g. [16, 28, 25].

Theorem 2 *If $M = M'$ then the function $x(t)'Mx(t)$ is strictly increasing with increasing t along any non-stationary trajectory $x(t)$ under both continuous-time (3) and discrete-time (4) replicator dynamics. Furthermore, any such trajectory converges to a stationary point. Finally, a vector $x \in S^n$ is asymptotically stable under (3) and (4) if and only if x is a strict local maximizer of $x'Mx$ on S^n.*

The previous result naturally suggests the use of replicator equations for approximating the MCP. In fact, let A_G be the (symmetric) adjacency matrix of graph G; by letting $M = A_G + \frac{1}{2}I$, the replicator dynamical system will iteratively maximize the Comtet objective function and eventually converge (with probability 1) to a local solution, which by virtue of Theorem 1, will then correspond to a characteristic vector of a maximal clique of G. One can also put $M = A_G$, in which case we obtain the Motzkin-Straus program, but due to the presence of spurious maximizers, solutions can only provide an approximation of the size of the maximum clique. The empirical results obtained in [12] over numerous DIMACS benchmark graphs are encouraging and prove the effectiveness of this algorithm. They also show that the approach based on the original (non-regularized) version of the Motzkin-Straus problem performs slightly better than its regularized counterpart (1), in terms of clique size. This may be intuitively explained by observing that, since all local maxima are strict, the landscape of the new objective function is certainly less flat than the one associated to the non-regularized version. This therefore enhances the tendency of replicator models to get stuck in inefficient local maxima.

3 Bounds for the annealing parameter

In this section, we establish bounds for the annealing parameter α related to the stability of x^S under the replicator dynamics. The first results hold for general symmetric matrices; we then specialize these findings to the case of adjacency matrices.

Proposition 3 *If $x \in S^n$ is a (local) maximizer of $x'(A+\alpha I)x$ over S^n and $S = \{i \in V : x_i > 0\}$, then necessarily*

$$\alpha \geq \gamma(x) = \max\{(Ax)_i - x'Ax : i \notin S\}/(x'x)\,. \tag{6}$$

Proof. Since x is a local maximizer of $x'(A+\alpha I)x$ on S^n, then has to satisfy the *Nash* equilibrium condition (see Theorem 2 of [12]) which, in this case, entails $(Ax)_i + \alpha x_i \leq x'Ax + \alpha x'x$ for all $i \notin S$ (note that equality has to hold if $i \in S$). But $i \notin S$ means $x_i = 0$ so that $\alpha \geq \gamma(x)$ follows readily. □

We move to the following result: a local maximizer x of both $x'Ax$ and $x'(A+\alpha I)x$ over S^n necessarily has to be a characteristic vector.

Proposition 4 *If $x \in S^n$ is a (local) maximizer of both $x'Ax$ and $x'(A+\alpha I)x$ over S^n for some $\alpha \neq 0$, then necessarily $x = x^S$ if $S = \{i \in V : x_i > 0\}$.*

Proof. From Theorem 2 of [12] we know that every local maximizer has to be a fixed point under the respective replicator dynamics. Hence for all $i \in S$ we have

$$[Ax]_i = x'Ax \quad \text{and also} \quad [Ax]_i + \alpha x_i = x'(A+\alpha I)x = x'Ax + \alpha x'x\,, \tag{7}$$

so that all positive coordinates of x have to be equal. Since they sum up to one, the result follows. □

Now we need some additional notation. First, denote by $e = (1, \cdots, 1)' \in \mathbb{R}^n$ and denote the hyperplane of all vectors the coordinates of which sum up to zero by

$$e^\perp = \{v \in \mathbb{R}^n : e'v = 0\}\,.$$

Given an arbitrary $n \times n$ matrix M, its action on the $(n-1)$-dimensional hyperplane $e^\perp$ can be fully described by $[Mv]_i = m_{i1}v_1 + \sum_{j>1} m_{ij}v_j = \sum_{j>1}[m_{ij} - m_{i1}]v_j$ for all $i = 1, \ldots, n$.

Similarly, if also $u \in e^{\perp}$, we get for the bilinear form

$$u'Mv = (-\sum_{i>1} u_i)[Mv]_1 + \sum_{i>1} u_i[Mv]_i = \widehat{u}'(M|e^{\perp})\widehat{v}$$

where $\widehat{u} = [u_i]_{i>1}$ is the truncated $(n-1)$-vector and $(M|e^{\perp})$ denotes the reduced $(n-1)\times(n-1)$ matrix

$$(M|e^{\perp}) = [m_{ij} - m_{1j} - m_{i1} + m_{11}]_{i,j>1}\,,$$

which is symmetric if M is symmetric.

We recall Theorem 5 of [11] according to which every local maximizer z of $x'Ax$ is maximizing this function over the whole face $\{y \in S^n : y_i = 0 \text{ if } z_i = 0\}$, and this face is contained in the basin of attraction of z under the replicator dynamics. This gives us an upper bound for the parameter α as follows:

Proposition 5 *Suppose for simplicity that $x = x^V = b$ is the barycenter of S^n, i.e., $b_i = \frac{1}{n}$ for all $i \in V$. If b is a global maximizer of $x'Ax$ on S^n, then $b \in S^n$ is also a (global) maximizer of $x'(A + \alpha I)x$ over S^n provided that*

$$\alpha \le \beta = \lambda_{\min}(-A|e^{\perp})\,, \tag{8}$$

where $\lambda_{\min}(-A|e^{\perp})$ is the smallest eigenvalue of the $(n-1)\times(n-1)$ matrix corresponding to the action of $-A$ on $e^{\perp}$.

Proof. The *Nash* equilibrium condition for b w.r.t. A implies $x'Ab = b'Ab$. But then $x'Ax - b'Ab = x'Ax - 2x'Ab + b'Ab = (b-x)'A(b-x) \le -\lambda_{\min}(-A|e^{\perp})(b-x)'(b-x)$. On the other hand, we also get $0 \le (b-x)'(b-x) = b'b - 2x'b + x'x = \frac{1}{n} - 2\frac{1}{n} + x'x = x'x - \frac{1}{n} = x'x - b'b$, whence

$$\begin{aligned} x'(A+\alpha I)x - b'(A+\alpha I)b &= x'Ax - b'Ab + \alpha(x'x - b'b) \\ &\le \alpha(x'x - b'b) - \lambda_{\min}(-A|e^{\perp})(b-x)'(b-x) \\ &= (\alpha - \beta)(x'x - b'b) \le 0 \end{aligned} \tag{9}$$

results provided $\alpha \le \beta$ holds, since the last expression in parentheses is always non-negative. Hence the result. □

Summarizing, we obtain an admissible range for our parameter:

Theorem 6 *Let $S \subseteq \{1, \cdots, n\}$, with $|S| = m$. Let $A_S = (a_{ij})_{i,j\in S}$ be the $m \times m$ submatrix of A corresponding to S, and $e_S = (1, \ldots, 1)' \in I\!R^m$. Denote by $\beta_S = \lambda_{\min}(-A_S|e_S^\perp)$.*
If $x = x^S \in S^n$ is a (local) maximizer of $x'Ax$ over S^n and $\alpha \in]\gamma(x^S), \beta_S[$, then x^S is also a local maximizer of $x'(A + \alpha I)x$ over S^n.
On the other hand, if $\alpha < \gamma(x^S)$, then x^S becomes an unstable fixed point of the replicator dynamics under $A+\alpha I$, and thus cannot be approached by an interior path under these dynamics with probability one.

Proof. From Theorem 2 of [12], the claimed assertion follows if we can establish local asymptotic stability of x^S under the replicator dynamics with, say, continuous time, and the matrix $A + \alpha I$. Now x^S lies in the relative interior of the face

$$F = \{x \in S^n : x_i = 0 \text{ for all } i \notin S\}$$

of the simplex, which in turn is also time-invariant under dynamics (3) and (4). As a consequence, we can decompose local stability analysis into the question of "internal" stability (concerning convergence of trajectories starting nearby within F) and, separately, "external" stability dealing with trajectories starting in S^n but off F. External stability is governed by the "external" eigenvalues of the linearization of this dynamics around x^S, which are given by the quantities (see [7], Lemma 21)

$$[(A + \alpha I)x^S]_i - (x^S)'(A + \alpha I)(x^S) = [Ax^S]_i - (x^S)'A(x^S) - \alpha(x^S)'(x^S) < 0$$

where negativity follows as in the proof of Proposition 3. On the other hand, internal stability of x^S follows from the optimality of x^S w.r.t. $x'(A + \alpha I)x$ on the face F as in Proposition 5, which is guaranteed by Theorem 5 of [11]. Hence the result for $\alpha > \gamma(x^S)$. The instability result follows by the same argumentation as in [13], Theorem 6: all starting points of trajectories converging to the non-asymptotically stable point x^S lie on the center-stable manifold [29, 30] which always is of codimension at least one. Hence a trajectory with a randomly chosen starting point will almost surely not converge to x^S. $\square$

A further result holds when A is the adjacency matrix of a graph G:

Theorem 7 *If $A = A_G$ is the adjacency matrix of graph G and S is a strictly maximal clique, then $]-1, 1[\ \subseteq\ ]\gamma(x^S), \beta_S[$.*

Proof. If $A = A_G$ and $x = x^S$ is the characteristic vector of a strictly maximal clique of size k, we know that $\gamma(x^S)$ cannot exceed -1, because of

$$k([A_G x^S]_i - (x^S)' A_G (x^S)\,) = d_S(i) - (k-1) \leq -1 \tag{10}$$

for all $i \notin S$ due to the definition of strict maximality of S . Here $d_S(i) = \sum_{j \in S} a_{ij}$ denotes the degree of vertex i w.r.t. S, i.e., the number of vertices of S connected to i. Moreover, in the simplifying hypothesis of Proposition 5, if $x^V = b$ is a global maximizer of $x'Ax$, then G is a complete graph, i.e. $A_G = ee' - I$, so that $-A$ acts on $e^\perp$ as the identity map, so that $\beta = 1$. Returning to the general situation where we have to replace V with S, we see that by analogy, $\beta_S = 1$ must hold. □

Theorem 8 *If $0 < \alpha < 1$, then the only strict local maximizers of $x'(A_G + \alpha I)x$ over S^n (i.e. the only attracting fixed points under the replicator dynamics with $A_G + \alpha I$) are characteristic vectors x^S where S is a maximal clique.*

Proof. We show that even every local maximizer $y \in S^n$ is a characteristic vector, by virtually the same proof as of Theorem 9 in [11]: to this end, put $S = \{i : y_i > 0\}$. First we show that the subgraph of G induced by S is complete. Indeed, suppose that for some $i, j \in S$ with $i \neq j$ we had $(i,j) \notin E$, i.e. $a_{ij} = 0$ would hold. Then for small $\delta > 0$, the point $x = y + \delta(e_i - e_j) \in S^n$ where e_i is the i-th standard basis vector. Straightforward calculations now yield

$$x'(A + \alpha I)x = y'(A + \alpha I)y + \alpha\delta^2 > y'(A + \alpha I)y\,,$$

a contradiction to the optimality of y. Hence with A_S and e_S as defined in Theorem 6 we get $A_S = e_S e_S' - I$. Now the Karush-Kuhn-Tucker conditions necessary for local optimality yield, in particular, $A_S y_S + \alpha y_S + \mu e_S = o$ for some $\mu \in \mathbb{R}$, which gives, using $e_S' y = 1$,

$$y_S = \frac{1+\mu}{1-\alpha} e_S\,,$$

which, again using $e_S' y = 1$, yields $y = x^S$. It remains to show that S is maximal. So suppose that there is a vertex $i \notin S$ such that $d_S(i) = |S|$. But then as in (10),

$$[(A_G + \alpha I)x^S]_i - (x^S)'(A_G + \alpha I)(x^S) \;=\; \frac{d_S(i) - |S| + 1 - \alpha}{|S|} > 0\,,$$

contradicting the *Nash* equilibrium property of x^S, which is ensured by local optimality of x^S (cf. Proposition 3). Hence S is a maximal clique. $\square$

For the case $-1 < \alpha < 0$ no general result has been proven, but examples can be provided in which new (spurious) local maxima emerge which are not characteristic vectors of any subset of vertices, and at the same time local solutions in the form of characteristic vectors disappear. In the next paragraph we study small examples which illustrate this point.

4 Dynamic analysis of two prototypical examples

In this section we investigate the dynamic behavior of the replicator dynamics of $A_G + \alpha I$ as α varies over the entire real line, for the two smallest possible prototypes of graphs of size 3.

Our first example, sketched in Figure 1, is a graph of size 3 with two maximum cliques of size 2 intersecting in vertex 3. Hence A_G has zero entries with the exception of $a_{13} = a_{31} = a_{23} = a_{32} = 1$. This is a frequently considered counterexample exhibiting spurious solutions to the Motzkin-Straus program [40]. For this simple graph $f_\alpha(x)$ is defined on the two-dimensional simplex S^3 so that we can actually plot it illustrating graphically the findings of the previous paragraphs.

More precisely, S^3 is a triangle spanned by the vertices $[1, 0, 0]'$, $[0, 1, 0]'$ and $[0, 0, 1]'$ contained in the plane described by the equation $x_1 + x_2 + x_3 = 1$. Our plots take this plane as their horizontal plane setting their origin in the vertex $[0, 0, 1]'$ of S^3. To remind the reader of this situation the plots of Figure 1 contain the S^3 triangle marked in gray; the third, vertical, axis of the plots report the values of $f_\alpha(x)$.

The plots of Figure 1 contain the basic Motzkin-Straus program and the regularizer term $x'x$, respectively; Figure 2 contain plots of $f_{\frac{1}{2}}(x)$ and of $f_{-\frac{1}{2}}(x)$ seen from a slightly different viewpoint. One can intuitively grasp what is happening by realizing that these two plots are obtained by the first one of Figure 1 when the regularizing term is respectively added or subtracted.

Let us now examine the figures in detail starting from the basic Motzkin-Straus program in Figure 1. In this example there are two maximum cliques $S = \{1, 3\}$ and $T = \{2, 3\}$, and their characteristic vectors $x^S = [\frac{1}{2}, 0, \frac{1}{2}]'$ and $x^T = [0, \frac{1}{2}, \frac{1}{2}]'$ give $f_0(x^S) = f_0(x^T) = \frac{1}{2}$. But more generally $f_0(x) = \frac{1}{2}$ for all $x = [\frac{1}{2} - b, b, \frac{1}{2}]'$ when $0 \le b \le \frac{1}{2}$. This example shows explicitly that

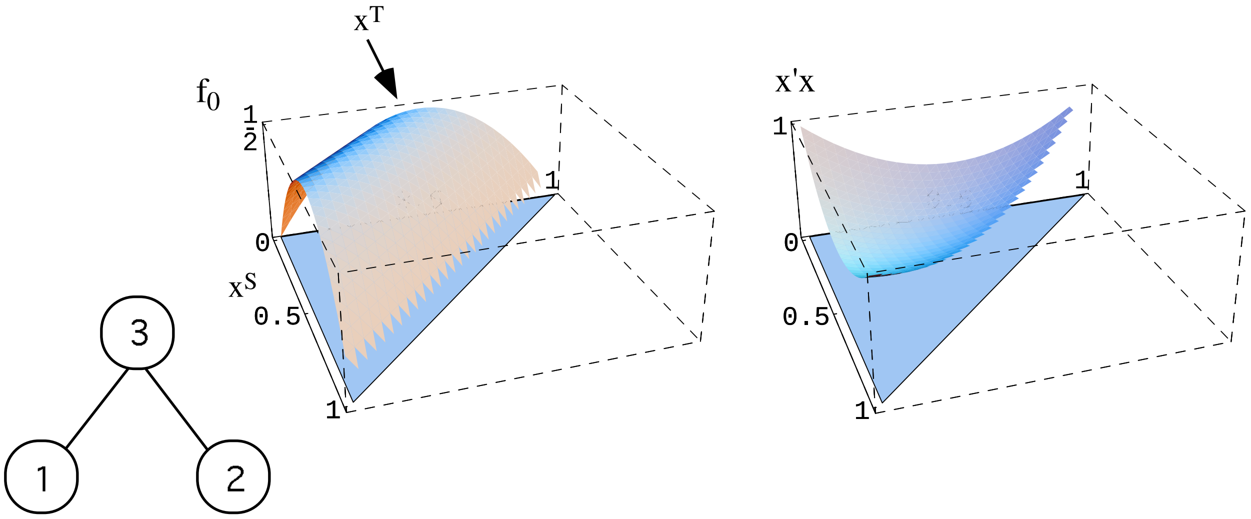

Figure 1: The Motzkin-Straus program $f_0(x)$ and $x'x$ when x varies over S^3

in the Motzkin-Straus program the global maximizer of $f_0(x)$ are not necessarily characteristic vectors of maximum cliques. It can be shown that this is just an instance of a more general property [45].

Figure 2 contains $f_{\frac{1}{2}}(x)$ and shows that in this case the only maxima correspond to the characteristic vectors of the maximum cliques while the second plot, containing $f_{-\frac{1}{2}}(x)$, shows that the isolated maximizer is an interior point not corresponding to any clique vector.

A more detailed analysis shows that there are three bifurcations at $\alpha = 0, 1, 2$. Since for the replicator dynamics on S^3 a complete classification of the replicator flow is available [9, 10], we refrain from repeating the phase portraits in pictures here, but rather refer to the numbers in the system used in the cited articles (see Fig. 6 in [9] and Fig. 1 in [10]). A prefixed minus sign $(-)$ symbolizes time reversal of the respective phase portrait (PP). In some cases, the PPs have to be rotated accordingly.

For $\alpha < 0$ the situation is essentially that of the second plot of Figure 2 and there is only the interior attractor

$$y_\alpha = [\frac{1-\alpha}{4-3\alpha}, \frac{1-\alpha}{4-3\alpha}, \frac{2-\alpha}{4-3\alpha}]' .$$

The flow is depicted as PP 7 in Fig. 6 of [9]. As α increases reaching the first bifurcation at $\alpha = 0$ the situation changes from that of the second plot of Figure 2 to that of the Motzkin-Straus program of Figure 1. The flow stops not only at the edge connecting vertices 1 and 2, but also along the trajectories joining y_α with the (former) saddle points x^S and x^T, so that we arrive at PP 1. If $\alpha \in]0, 1[$ the situation is essentially that of the first plot of Figure 2,

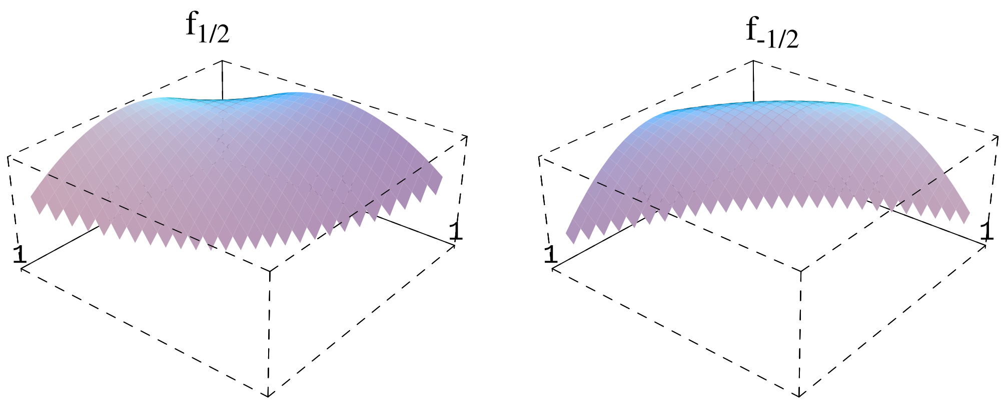

Figure 2: $f_{\frac{1}{2}}(x)$ and $f_{-\frac{1}{2}}(x)$ on S^3

and we have the picture of PP 8, which renders both x^S and x^T as (local) attractors, and y_α as a saddle point wandering towards $x^{\{3\}}$ as $\alpha \nearrow 1$. This completely breaks down if $\alpha = 1$ where, again, the flow is stopped at two edges: PP 20 emerges. If α is increased further, only the vertices are attracting with the occurrence of y_α as an interior repellor if α exceeds 2: we get PP -35 for $\alpha \in]1, 2]$ and PP -7 if $\alpha > 2$. The regularizing term of Figure 1 substantially depicts the situation after the last bifurcation at $\alpha = 2$.

In the next example, G consists of three vertices with only two of them connected, so A_G has only zero entries with the exception of $a_{13} = 1$. Hence $S = \{1, 3\}$ is the unique maximum clique. In this case, replicator dynamics undergoes a simple exchange-of-stability bifurcation as α passes through $\alpha = -1$, and a more dramatic, but similar phenomenon occurs as $\alpha = 0$ where the flow on two edges is reversed simultaneously: indeed, for $\alpha < -1$ we obtain, again, PP No. 7 with interior global attractor of the form

$$x_\alpha = [\frac{\alpha}{3\alpha+1}, \frac{\alpha}{3\alpha+1}, \frac{\alpha+1}{3\alpha+1}]',$$

(observe that x_α approaches $[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]'$ as $\alpha \searrow -\infty$ while $x_\alpha \to x^S = [\frac{1}{2}, \frac{1}{2}, 0]'$ as $\alpha \nearrow -1$). If α now increases, this stable fixed point remains at x^S. Indeed, the PP is for $\alpha \in [-1, 0[$ qualitatively the same as PP 35 while for $\alpha = 0$ we obtain the PP -20. Still x^S is the global attractor. If α is further increased to positive numbers (e.g. $\alpha = \frac{1}{2}$), then local stability of x^S is retained, but there emerges a second local attractor $x^{\{3\}}$ together with an interior saddle point, again at x_α as above, and the PP -8 results. If $\alpha = 1$, stability of $x^{\{3\}}$ is retained, but the flow at the edge containing x^S stops: PP -1 depicts the situation, which in some sense corresponds to the occurrence of spurious solutions in the previous example (emerging at $\alpha = 0$ there). Finally, if $\alpha > 1$, we arrive again at PP -7 where x_α is now a repellor and all vertices become attractors. This is in accordance of the theory as in Theorem 8.

5 The annealed replication heuristic

As discussed previously, the major drawback of replicator equations is their inherent inability to escape from local maximizers of the objective function. Theorem 6 provides us with an immediate strategy to avoid unwanted local solutions, i.e., maximal cliques which are not maximum. Suppose that S is a maximal clique in G that we want to avoid. By letting $\alpha < \gamma(x^S)$, its characteristic vector x^S becomes an unstable stationary point of the replicator dynamics under f_α, and thus will not be approached by any interior trajectory. Of course, the problem is to obtain a reasonable estimate for $\gamma(x^S)$ without knowing S in advance. Furthermore, if $\alpha \leq 0$, it may well happen that the process converges to a vector which does not represent a clique (see below).

Since we are concerned with the maximum clique problem,

$$\gamma(x^S) = \gamma_S = \max_{i \notin S} d_S(i) - |S| + 1\,. \tag{11}$$

As already noted in (10), $\gamma_S \leq -1$ if S is strictly maximal while $\gamma_S = 0$ if S is not strictly maximal. In both cases, $\gamma_S \geq 1 - |S|$ with equality attained if S is isolated in G. So if one wants to avoid cliques with size $|S| \leq m$, one could simply run the algorithm with $\alpha < 1 - m \leq 1 - |S| \leq \gamma_S \leq 0$, and if there is a clique T such that still $\gamma_T < \alpha$ holds, there is a (more or less justified) hope to obtain in the limit x^T, which yields *automatically* a larger maximal clique T.

Unfortunately, two other cases could occur:

(a) no other clique T satisfies $\gamma_T < \alpha$, i.e., α has too a large absolute value

(b) even if there is such a clique, other attractors could emerge which are not characteristic vectors of a clique. (Note that this is excluded if $\alpha > 0$ by Theorem 8.)

The proper choice of the parameter α is therefore a trade-off between the desire to remove unwanted maximal cliques and the emergence of spurious solutions.

Instead of keeping the value of α fixed, our approach is to start with a sufficiently large negative α and adaptively decrease it during the optimization process, in much the same spirit as the simulated annealing procedure [32]. Of course, in this case the annealing parameter has no interpretation in terms of a hypothetical temperature, and the resulting algorithm is completely deterministic. The rationale behind this idea is that for values of α with a proper large absolute value only the characteristic vectors of large maximal cliques will be stable attractive points for the replicator dynamics, presumably together with a set of spurious solutions. As the value of α is reduced, spurious solutions disappear and at the same time (characteristic vectors of) smaller maximal cliques become stable. We expect that at the beginning of the annealing process the dynamics is attracted toward "promising" regions, and the search is further refined as the annealing parameter decreases. This picture seems to be confirmed by the experimental results presented in the next section, and is also intuitively clear by the simple illustrative examples shown in Section 4. In summary, the proposed algorithm is as follows:

1. Start with a sufficiently large positive α;
2. Let b be the barycenter of S^n and set $x = b$;
3. Run the replicator dynamics starting from x, under $A - \alpha I$ until convergence and let x be the converged point;
4. Unless a stopping condition is met, decrease α and goto 3;
5. Select $\overline{\alpha}$ with $0 < \overline{\alpha} < 1$, run the replicator dynamics starting from x under $A + \overline{\alpha} I$ until convergence, and extract a maximal clique from the converged solution.

The last step in the algorithm is necessary if we want to extract also the vertices comprising the clique found, as shown in Theorem 8. A typical choice is to set $\overline{\alpha} = \frac{1}{2}$ in this case.

It is clear that for the algorithm to work, we need to select an appropriate "annealing" strategy. To this end, one could employ the following heuristics: suppose for the moment that the underlying graph is a random one in the sense that edges are generated independently of each other with a certain equal probability q (in applications, q could be replaced with $|E|/\binom{n}{2}$, the actual density). Suppose S is an unwanted clique of size m. Take $\delta > 0$ small, say 0.01, and consider a lower bound which is exceeded with probability $1 - \delta$:

Theorem 9 *Under the random graph model consider a clique S of size $|S| = m$, put $\nu = \frac{1}{2(n-m)}$ and denote by $\overline{\gamma}_m$ the following lower bound for $\gamma(x^S)$:*

$$\overline{\gamma}_m = 1 - (1-q)m - \sqrt{mq(1-q)}\,\delta^\nu\,. \tag{12}$$

Then

$$\mathbb{P}(\gamma(x^S) \leq \overline{\gamma}_m) \leq \delta\,.$$

Moreover, $\overline{\gamma}_m$ exceeds $1-m$ for all $m \geq m_{q,\delta}$ where

$$m_{q,\delta} = \frac{1-q}{q}\,\sqrt[n]{\delta}\,. \tag{13}$$

Proof. Since $\gamma(x^S) = \max_{i\notin S} d_S(i) - m + 1$, and since for different $i \neq j$, the variates $d_S(i)$ and $d_S(j)$ are stochastically independent in the random graph model, we first get the identity

$$\mathbb{P}(\gamma(x^S) \leq \overline{\gamma}_m) = [\mathbb{P}(d_S(i) \leq \overline{\gamma}_m + m - 1)]^{n-m}\,.$$

Next observe that the expected value and variance of $d_S(i)$ is, according to the Binomial Law,

$$\mathbb{E}d_S(i) = mq \quad \text{and} \quad \mathrm{Var}\ d_S(i) = mq(1-q)\,.$$

Hence Čebyšev's inequality gives

$$\mathbb{P}(|d_S(i) - mq| \geq \varepsilon) \leq mq(1-q)/\varepsilon^2\,,$$

which entails, putting $\varepsilon = 1-(1-q)m - \overline{\gamma}_m > 0$,

$$\begin{aligned} \mathbb{P}(d_S(i) \leq \overline{\gamma}_m + m - 1) &= \mathbb{P}(d_S(i) - mq \leq \overline{\gamma}_m + (1-q)m - 1) \\ &= \mathbb{P}(d_S(i) - mq \leq -\varepsilon) \\ &\leq \mathbb{P}(|d_S(i) - mq| \leq \varepsilon) \leq mq(1-q)\varepsilon^{-2}\,. \end{aligned} \tag{14}$$

This gives an lower bound $\overline{\gamma}_m$ for γ_S which is exceeded with a probability of least $1-\delta$ as follows:

$$\begin{aligned} \mathbb{P}(\gamma(x^S) \leq \overline{\gamma}_m) &= [\mathbb{P}(d_S(i) < \overline{\gamma}_m + m - 1)]^{n-m} \\ &\leq [mq(1-q)\varepsilon^{-2}]^{n-m} = \delta \end{aligned} \tag{15}$$

provided $1-(1-q)m-\overline{\gamma}_m = \varepsilon = \sqrt{mq(1-q)}\ \sqrt[2(n-m)]{\delta}$, which yields (12). Since $\nu \geq \frac{1}{2n}$ by definition, obviously $\delta^{2\nu} \leq \delta^{\frac{1}{n}}$. On the other hand, $\overline{\gamma}_m > 1-m$ if and only if $qm - \sqrt{mq(1-q)}\delta^\nu > 0$, which is equivalent to $m > \frac{1-q}{q}\delta^{2\nu}$. Hence $m > m_{q,\delta} \geq \frac{1-q}{q}\delta^{2\nu}$ yields $\overline{\gamma}_m > 1-m$. □

Since $m_{q,\delta} \leq 10$ if $q \geq 0.1$ for all $\delta > 0$, the previously obtained hard lower bound $1 - m$ is relaxed by $\overline{\gamma}_m$ in almost all important applications. Moreover the bound $\overline{\gamma}_m$ decreases with increasing m provided that

$$\frac{1}{\sqrt{m}}[1 + \frac{m}{(n-m)^2} \log \delta] > -\frac{2}{\delta^\nu}\sqrt{\frac{1-q}{q}}$$

holds, and this is true for many important cases in practice: indeed, observe that the latter inequality necessarily holds if the expression in brackets is positive, which is true, e.g. for $\delta = 0.01$, whenever $6m \leq (n-m)^2$. Thus it makes sense to use $\overline{\gamma}_m$ as a heuristic proxy for the lower bound of $\gamma(x^S)$, to avoid being attracted by a clique of size m.

Furthermore, a well-known result due to Matula [36] accurately predicts the size of the maximum clique in random graphs with sufficiently many vertices. Let

$$M(n,q) = 2\log_{1/q} n - 2\log_{1/q}\log_{1/q} n + 2\log_{1/q}\frac{e}{2} + 1 \ . \tag{16}$$

Matula proved that, as $n \to \infty$, the size of the maximum clique in an n-vertex q-density random graph is either $\lfloor M(n,q) \rfloor$ or $\lceil M(n,q) \rceil$ with probability tending to 1.

The previous results suggest us a sort of "two-level" annealing strategy: the level of clique size, which in turn induces that of the "actual" annealing parameter. More precisely, if we do not have any a priori information about the expected size of the maximum clique, we can use Matula's formula $M(n,q)$ to have an initial (more or less accurate) estimate of it. Let $m = \lceil M(n,q) \rceil$; by setting the initial value for α (step 1 of our algorithm) at some intermediate value between $\overline{\gamma}_m$ and $\overline{\gamma}_{m-1}$, e.g, $\alpha = (\overline{\gamma}_m + \overline{\gamma}_{m-1})/2$, we expect that only the characteristic vectors of maximal cliques having size m will survive in f_α, together with many spurious solutions. After the initial cycle, we decrease m, recalculate $\overline{\gamma}_m$ and $\overline{\gamma}_{m-1}$ and update $\alpha = (\overline{\gamma}_m + \overline{\gamma}_{m-1})/2$ in step 4 as in the previous step. The whole process is iterated until either m reaches 1 or α becomes smaller than a fixed threshold.

6 Experimental results

To assess the effectiveness of the proposed heuristic, extensive simulations were carried out over a selection of DIMACS graphs [27], which represent a standard benchmark for clique finding algorithms.[1] The experiments were conducted using the discrete-time version (4) of

[1] We did not consider graphs were the plain Comtet algorithm already yields the maximum clique, e.g., the "c-fat" family [12]. Also, a few very large and dense graphs were excluded because of the excessively high computational cost required.

the replicator equations. The code was written in the C programming language and run on a Digital AlphaStation Series 200 (no attempt was made to optimize the code). For each graph considered, the proposed algorithm was run by using the two-level annealing schedule described at the end of the previous section. For each internal cycle (step 3), the replicator algorithm was iterated until the (squared) distance between two successive states became smaller than 10^{-10}. At the final cycle (i.e., step 5), the parameter α was set to $\frac{1}{2}$, and the replicator dynamics was stopped when either a maximal clique (i.e., a local maximizer of $f_{\frac{1}{2}}$ on S^n) was found or the distance between two successive points was smaller than a fixed threshold, which was set to $n10^{-15}$ (n being the number of vertices of the graph at hand). In the latter case the converged vector was randomly perturbed, and the algorithm restarted from the perturbed point. Because of the one-to-one correspondence between local maximizers and maximal cliques (see Theorem 8) this situation corresponds to convergence to a saddle point.

In order to assess the relative merits of the proposed heuristic we compare our results with those obtained using a fixed positive α, i.e., with no annealing strategy), which corresponds to the original quadratic program proposed by Comtet and recently studied by Bomze *et al.* [11, 12]. In addition, our results were compared with those reported by Gibbons *et al.* in [22] who proposed a continuous-based heuristic (CBH) also based on a parametrization (completely different from ours) of the Motzkin-Straus program.

The results of our experiments are summarized in Table 1 and 2, which contain a row for each DIMACS graphs considered. The columns labeled Graph, Vertices, and Dens. represent the name of the corresponding graph, the number of its vertices and its density, respectively. The column Max Clique, contains the size of the maximum clique when known, or a lower bound for it (this information is already available in the file containing the graph). The columns AR, Comtet and CBH, contain the size of the clique found using the proposed annealed replication heuristic, the replicator dynamics under the plain Comtet program (these results are taken from [12]), and Gibbons *et al.*'s CBH algorithm [22]. Finally, the column labeled Time contain the CPU time required by the process to provide the final solution.

As can be seen, the results are very encouraging. In fact, in all but one case we obtained larger cliques than Comtet did, the exception being `san200_0.9_3` where AR returned a clique size equal to 31, while Comtet obtained 32. Note that in this case, CBH returned a clique size which was worse than both Comtet and AR, i.e. 30. In many cases, we obtained the

Graph	*Vertices*	*Dens.*	*Max Clique*	Clique Size Obtained *AR*	*Comtet*	*CBH*	Time *(secs)*
brock200_1	200	0.745	21	19	17	20	167.79
brock200_2	200	0.496	12	10	8	12	97.33
brock200_3	200	0.605	15	13	9	14	124.44
brock200_4	200	0.658	17	14	12	16	150.74
brock400_1	400	0.748	27	20	21	23	906.26
brock400_2	400	0.749	29	23	20	24	752.69
brock400_3	400	0.748	31	23	18	23	554.45
brock400_4	400	0.749	33	23	19	24	937.84
brock800_1	800	0.649	23	18	16	20	3323.31
brock800_2	800	0.651	24	18	15	19	3175.44
brock800_3	800	0.649	25	19	16	20	2697.56
brock800_4	800	0.650	26	19	15	19	3181.74
san1000	1000	0.501	15	8	8	8	1824.60
san200_0.7_1	200	0.700	30	15	15	15	39.66
san200_0.7_2	200	0.700	18	12	12	12	40.11
san200_0.9_1	200	0.900	70	45	45	46	106.29
san200_0.9_2	200	0.900	60	39	36	36	56.03
san200_0.9_3	200	0.900	44	31	32	30	98.76
san400_0.5_1	400	0.500	13	7	7	8	156.77
san400_0.7_1	400	0.700	40	20	20	20	232.39
san400_0.7_2	400	0.700	30	15	15	15	230.72
san400_0.7_3	400	0.700	22	12	12	14	194.12
san400_0.9_1	400	0.900	100	50	40	50	425.88
sanr200_0.7	200	0.700	18	16	14	18	131.12
sanr200_0.9	200	0.900	≥ 42	41	37	41	158.41
sanr400_0.5	400	0.900	13	13	11	12	269.64
sanr400_0.7	400	0.700	≥ 21	21	18	20	838.30

Table 1: Results on DIMACS benchmark graphs (part I)

Graph	*Vertices*	*Dens.*	*Max Clique*	Clique Size Obtained *AR*	*Comtet*	*CBH*	Time *(secs)*
MANN_a9	45	0.927	16	16	12	16	0.833
MANN_a27	378	0.990	126	117	117	121	6807.81
p_hat300-1	300	0.244	8	8	6	8	107.63
p_hat300-2	300	0.489	25	25	22	25	301.69
p_hat300-3	300	0.744	36	35	32	36	3221.27
p_hat500-1	500	0.253	9	9	8	9	335.83
p_hat500-2	500	0.505	36	36	33	35	893.21
p_hat500-3	500	0.752	$\geq$49	47	47	49	1729.50
p_hat700-1	700	0.249	11	9	7	11	739.92
p_hat700-2	700	0.498	44	41	43	44	1893.96
p_hat700-3	700	0.748	$\geq$62	59	57	60	2582.18
p_hat1000-1	1000	0.245	10	10	8	10	1965.47
p_hat1000-2	1000	0.490	$\geq$46	44	42	46	3010.43
p_hat1000-3	1000	0.744	$\geq$65	62	61	65	7288.22
p_hat1500-1	1500	0.253	12	10	9	11	4100.54
p_hat1500-2	1500	0.506	$\geq$63	**64**	62	63	8598.91
p_hat1500-3	1500	0.754	$\geq$94	91	89	94	16251.77
keller4	171	0.649	11	8	7	10	34.85
keller5	776	0.751	27	16	15	21	610.16

Table 2: Results on DIMACS benchmark graphs (part II)

same results as CBH and in a few examples we returned better solutions, e.g., p_hat1500-2, san200_0.9_2, sanr400_0.5. In one case (p_hat1500-2) we found a clique bigger than the known lower bound. It is worth noting that the Sanchis graphs (the "san" family) turned out to be very hard for Motzkin-Straus-based optimization algorithms since neither of the three heuristics found good results.

As far as the CPU time is concerned, it is clear that our algorithm turns out to be computationally more expensive than plain replicator dynamics on fixed positive α (see [12] for comparison) because the latter is simply the final step of our heuristic (step 5). Moreover, AR is slower than CBH [22] which in turn may have serious memory allocation problems. However, we note that the continuous-time version (3) of replicator equations can naturally be mapped onto hardware circuitry [47], thereby making the whole algorithm particularly amenable to parallel, distributed implementations.

¿From the results obtained, it can be concluded that the proposed annealed replication heuristic does a good job at finding large cliques, and systematically beats the plain Comtet approach, where no annealing strategy is used. Moreover, it should be pointed out that the annealing schedule adopted is entirely based on the assumption that the graphs at hand are random; clearly, DIMACS graphs can hardly be said "random," but nevertheless the heuristic worked remarkably well. Of course, better annealing strategies could be devised if we knew something about the underlying structure of the graphs, but in absence of this kind of information the random graph assumption seems to be sufficiently robust.

7 Conclusions

We have presented a new heuristic for approximating the maximum clique problem. The approach is centered around an attractive characterization of the problem due to Motzkin and Straus, which allows us to formulate it as a linearly constrained quadratic maximization program. Specifically, we have introduced a control parameter α and studied the properties of the objective function as α varies. We have shown that when α is positive all the properties enjoyed by the standard Comtet approach [11] hold true; specifically, in this case a one-to-one correspondence between local/global maximizers in the continuous space and local/global solutions in the discrete space exists. For negative α's an interesting picture emerges: as the absolute value of α becomes larger, local maximizers corresponding to maximal cliques disappear. We have derived bounds on the parameter α which affect the stability of these solutions. These results have suggested the annealed replication heuristic, which consists of starting from a large negative α and then properly reducing it during the optimization process. For each value of α standard replicator equations are run in order to obtain local solutions of the corresponding objective function. The rationale behind this idea is that for values of α with a proper large absolute value only local solutions corresponding to large maximal cliques will survive, together with various spurious maximizers. As the value of α is reduced, spurious solutions disappear and smaller maximal cliques become stable. An annealing schedule is proposed which is based on the assumption that the graphs being considered are random. Experiments conducted over several DIMACS benchmark graphs confirm the effectiveness of the proposed approach and the robustness of the annealing strategy. The overall conclusion is that the annealing procedure *does* help to avoid inefficient local solutions, by initially driving the dynamics towards promising regions in state space, and then refining the search as the annealing parameter is decreased.

References

[1] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy, "Proof verification and the hardness of approximation problems," in *Proc. 33rd Ann. Symp. Found. Comput. Sci.*, Pittsburgh, PA, 1992, pp. 14–23.

[2] — and S. Safra, "Probabilistic checking of proofs: A new characterization of NP," in *Proc. 33rd Ann. Symp. Found. Comput. Sci.*, Pittsburgh, PA, 1992, pp. 2–13.

[3] M. Bellare, S. Goldwasser, C. Lund, and A. Russell, "Efficient probabilistically checkable proofs and application to approximation," in *Proc. 25th Ann. ACM Symp. Theory Comput.*, San Diego, CA, 1993, pp. 294–304.

[4] — , — , and M. Sudan, "Free bits, PCPs and non-approximability—Towards tight results," in *Proc. 36th Ann. Symp. Found. Comput. Sci.*, Milwaukee, WI, 1995, pp. 422–431.

[5] L. E. Baum and J. A. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bull. Amer. Math. Soc.*, vol. 73, pp. 360–363, 1967.

[6] — and G. R. Sell, "Growth transformations for functions on manifolds," *Pacif. J. Math.*, vol. 27, no. 2, pp. 211–227, 1968.

[7] I. M. Bomze, "Non-cooperative two-person games in biology: a classification," *Int. J. Game Theo.*, vol. 15, pp. 31–57, 1986.

[8] — , "Cross entropy minimization in uninvadable states of complex populations," *J. Math. Biol.*, vol. 30, pp. 73–87, 1991.

[9] — , "Lotka-Volterra equation and replicator dynamics: A two-dimensional classification," *Biol. Cybern.*, vol. 48, pp. 201–211, 1983.

[10] — , "Lotka-Volterra equation and replicator dynamics: New issues in classification," *Biol. Cybern.*, vol. 72, pp. 447–453, 1995.

[11] — , "Evolution towards the maximum clique," *J. Global Optim.*, vol. 10, pp. 143–164, 1997.

[12] — , M. Pelillo, and R. Giacomini, "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*. Kluwer, Dordrecht, 1997, pp. 95–108.

[13] — and V. Stix, "Genetic engineering via negative fitness: evolutionary dynamics for global optimization," to appear in *Annals of O.R.*

[14] M. Budinich, "Bounds on the maximum clique of a graph," submitted (see `http://www.ts.infn.it/~mbh/MC_Bounds.ps.Z`).

[15] L. Comtet, *Advanced Combinatorics*. Reidel, Dordrecht, 1974.

[16] J. F. Crow and M. Kimura, *An Introduction to Population Genetics Theory*. Harper & Row, New York, 1970.

[17] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy, "Approximating clique is almost NP-complete," in *Proc. 32nd Ann. Symp. Found. Comput. Sci.*, San Juan, Puerto Rico, 1991, pp. 2–12.

[18] — , — , — , — , —, "Interactive proofs and the hardness of approximating clique," *J. ACM*, vol. 43, no. 2, pp. 268–292, 1996.

[19] R. A. Fisher, *The Genetical Theory of Natural Selection*. Clarendon Press, Oxford, 1930.

[20] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.

[21] A. H. Gee and R. W. Prager, "Polyhedral combinatorics and neural networks," *Neural Computation*, vol. 6, pp. 161–180, 1994.

[22] L. E. Gibbons, D. W. Hearn, and P. M. Pardalos, "A continuous based heuristic for the maximum clique problem," In: D. S. Johnson and M. Trick (Eds.), *Cliques, Coloring, and Satisfiability—Second DIMACS Implementation Challenge.* American Mathematical Society, Providence, RI, 1996, pp. 103–124.

[23] — , — , — , and M. V. Ramana, "Continuous characterizations of the maximum clique problem," *Math. Oper. Res.*, vol. 22, no. 3, pp. 754–768, 1997.

[24] J. Hrastad, "Clique is hard to approximate within $n^{1-\epsilon}$," in *Proc. 37th Ann. Symp. Found. Comput. Sci.*, 1996, pp. 627–636.

[25] J. Hofbauer and K. Sigmund, *The Theory of Evolution and Dynamical Systems.* Cambridge University Press, Cambridge, UK, 1988.

[26] R. Horst, P. M. Pardalos, and N. V. Thoai, *Introduction to Global Optimization.* Kluwer Academic Publishers, Amsterdam, The Netherlands, 1995.

[27] D.S. Johnson and M.A. Trick (Eds.), *Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol 26, American Mathematical Society, Providence, RI, 1996 (see also `http://dimacs.rutgers.edu/Volumes/Vol26.html`).

[28] S. Karlin, "Mathematical models, problems and controversies of evolutionary theory," *Bull. Amer. Math. Soc.*, vol. 10, pp. 221–273, 1984.

[29] A. Kelley, "The stable, center-stable, center, center-unstable, unstable manifolds," *J. Differential Equations* vol. 3, pp. 546–570, 1967.

[30] — , "Stability of the center-stable manifold," *J. Math. Anal. Appl.* vol. 18, 336–344, 1967.

[31] M. Kimura, "On the change of population fitness by natural selection," *Heredity*, vol. 12, pp. 145–167, 1958.

[32] S. Kirkpatrick, C.D. Gelatt Jr., and M.P. Vecchi, "Optimization by Simulated Annealing," *Science* vol. 220(4598), pp. 671–679, 1983.

[33] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, "An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition," *Bell Syst. Tech. J.*, vol. 62, pp. 1035–1074, 1983.

[34] V. Losert and E. Akin, "Dynamics of games and genes: discrete versus continuous time," *J. Math. Biol.*, vol. 17, pp. 241–251, 1983.

[35] Yu. Lyubich, G. D. Maistrowskii, and Yu. G. Ol'khovskii, "Selection-induced convergence to equilibrium in a single-locus autosomal population," *Problems of Information Transmission*, vol. 16, pp. 66–75, 1980.

[36] D.W. Matula, "The Largest Clique Size in a Random Graph," *Technical Report CS 7608*, Department of Computer Science, Southern Methodist University, 1976.

[37] T. S. Motzkin and E. G. Straus, "Maxima for graphs and a new proof of a theorem of Turán," *Canad. J. Math.*, vol. 17, pp. 533–540, 1965.

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

[39] P. M. Pardalos, "Continuous approaches to discrete optimization problems," In: G. Di Pillo and F. Giannessi (Eds.), *Nonlinear Optimization and Applications.* Plenum Press, New York, 1996, pp. 313–328.

[40] — and A. T. Phillips, "A global optimization approach for solving the maximum clique problem," *Int. J. Computer Math.*, vol. 33, pp. 209–216, 1990.

[41] — and J. Xue, "The maximum clique problem," *J. Global Optim.*, vol. 4, pp. 301–328, 1994.

[42] M. Pelillo, "Relaxation labeling networks for the maximum clique problem," *J. Artif. Neural Networks*, vol. 2, pp. 313–328, 1995.

[43] — , "The dynamics of nonlinear relaxation labeling processes," *J. Math. Imaging Vision*, vol. 7, no. 4, pp. 309–323, 1997.

[44] — and I. M. Bomze, "Parallelizable evolutionary dynamics principle for solving the maximum clique problem," In: H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature IV*, Springer-Verlag, Berlin, 1996, pp. 676–685.

[45] — and A. Jagota, "Feasible and infeasible maxima in a quadratic program for maximum clique," *J. Artif. Neural Networks*, vol. 2, pp. 411–419, 1995.

[46] P. Taylor and L. Jonker, "Evolutionarily stable strategies and game dynamics," *Math. Biosci.*, vol. 40, pp. 145–156, 1978.

[47] A. Torsello and M. Pelillo, "Continuous-time relaxation labeling processes," *Technical Report*, Dipartimento di Matematica Applicata e Informatica, Università Ca' Foscari di Venezia, Italy, March 1998.

[48] J. W. Weibull, *Evolutionary Game Theory.* MIT Press, Cambridge, MA, 1995.