Replicator Equations, Maximal Cliques, and Graph Isomorphism *

Marcello Pelillo
Dipartimento di Informatica
Università Ca' Foscari di Venezia
Via Torino 155, 30172 Venezia Mestre, Italy
E-mail: pelillo@dsi.unive.it

* *Neural Computation* (accepted for publication)

Abstract

We present a new energy-minimization framework for the graph isomorphism problem which is based on an equivalent maximum clique formulation. The approach is centered around a fundamental result proved by Motzkin and Straus in the mid-1960s, and recently expanded in various ways, which allows us to formulate the maximum clique problem in terms of a standard quadratic program. The attractive feature of this formulation is that a clear one-to-one correspondence exists between the solutions of the quadratic program and those in the original, combinatorial problem. To solve the program we use the so-called "replicator" equations, a class of straightforward continuous- and discrete-time dynamical systems developed in various branches of theoretical biology. We show how, despite their inherent inability to escape from local solutions, they nevertheless provide experimental results which are competitive with those obtained using more elaborate mean-field annealing heuristics.

1 Introduction

The graph isomorphism problem is one of those few combinatorial optimization problems which still resist any computational complexity characterization (Garey and Johnson, 1979; Johnson, 1988). Despite decades of active research, no polynomial-time algorithm for it has yet been found. At the same time, while clearly belonging to NP, no proof has been provided that it is NP-complete. Indeed, there is strong evidence that this cannot be the case, for otherwise the polynomial hierarchy would collapse (Boppana, Hastad and Zachos, 1987; Schöning, 1988). The current belief is that the problem lies strictly between the P and NP-complete classes.

Because of its theoretical as well as practical importance, the problem has attracted much attention in the neural network community, and various powerful heuristics have been developed (Kree and Zippelius, 1988; Gold and Rangarajan, 1996; Mjolsness, Gindi and Anandan, 1989; Rangarajan, Gold and Mjolsness, 1996; Rangarajan and Mjolsness, 1996; Simić, 1991). Following Hopfield and Tank's seminal work (Hopfield and Tank, 1985) the customary approach has been to derive a (continuous) energy function in such a way that solutions of the original, discrete problem map onto minimizers of the function in a continuous domain. For graph isomorphism the continuous domain usually corresponds to the unit hypercube and the energy function is quadratic. The energy is then minimized using an appropriate dynamical system and, after convergence, a solution to the discrete problem is recovered from the minimizer thus found. Almost invariably, the minimization algorithms developed so far incorporate techniques borrowed from statistical mechanics, in particular mean field theory, which allow one to escape from poor local solutions.

Early formulations suffer from the lack of a precise characterization of the local/global minimizers of the continuous energy function in terms of the solutions of the discrete problem, which are usually in the form of a permutation matrix. In other words, while the solutions of the original problem correspond (by construction) to solutions of its continuous counterpart, the inverse is not necessarily true. Recently, however, Yuille and Kosowsky (1994) have shown that by adding a certain term to the quadratic objective, minimizers in the unit hypercube can only lie at the vertices, thereby overcoming this drawback. Their formulation has been successfully employed in conjunction with double-normalization and Lagrangian decomposition methods (Rangarajan, Gold and Mjolsness, 1996; Rangarajan and Mjolsness, 1996). An additional remark on standard neural network models for graph isomorphism is that it is not clear how to interpret the solutions of the continuous problem when the graphs being matched are not isomorphic. In this case, in fact, there is no permutation matrix which solves the problem, and yet there will be minima in continuous space since the domain is compact and the function being minimized is continuous. Even though this issue is more closely related to the subgraph isomorphism problem (which is known to be computationally intractable), it would be desirable for a graph isomorphism algorithm to always return "meaningful" solutions.

In this paper, we develop a new energy-minimization framework for graph isomorphism which is based on the idea of reducing it to the maximum clique problem, another well-known combinatorial optimization problem (Pardalos and Xue, 1994). Central to our approach is a powerful result originally proved by Motzkin and Straus (1965), and recently extended in various ways (Bomze, 1997; Gibbons, Hearn and Pardalos, 1996; Gibbons et al., 1997; Pelillo and Jagota, 1995), which allows us to formulate the maximum clique problem in terms of an indefinite quadratic program. In the proposed formulation an elegant one-to-one correspondence exists between the solutions of the quadratic program and those of the original problem. We also present a class of straightforward continuous- and discrete-time dynamical systems known in mathematical biology as *replicator equations*, and show how, owing to their properties, they provide a natural and useful heuristic for solving the Motzkin-Straus program, and hence the graph isomorphism problem.

It may be argued that trying to solve the graph isomorphism problem by reducing it to the maximum clique problem is an altogether inappropriate choice. In contrast to graph isomorphism, in fact, the problem of finding just the cardinality of the maximum clique in a graph is known to be NP-complete and, according to recent theoretical results, so is the problem of approximating it within a certain tolerance (Arora et al., 1992; Bellare, Goldwasser and Sudan, 1995; Hastad, 1996).[1] The experimental results presented in this paper, however, seem to contradict this claim. By using simple relaxation equations which are inherently unable to avoid local optima, we get results which compare favorably with those obtained using state-of-the-art sophisticated deterministic annealing algorithms which, by contrast, are explicitly designed to escape from local solutions. This suggests therefore that the proposed Motzkin-Straus formulation is a promising framework within which to develop powerful graph isomorphism heuristics.

The outline of the paper is as follows. Section 2 presents the quadratic programming formulation for graph isomorphism derived from the Motzkin-Straus theorem. In Section 3 we introduce the replicator equations, discuss their fundamental dynamical properties, and present the experimental results obtained over hundreds of 100-vertex graphs of various connectivities. In Section 4, an exponential replicator dynamics is presented which turns out to be dramatically faster and more accurate than the classical model. Finally, Section 5 concludes the paper.

2 A quadratic programming formulation for graph isomorphism

2.1 Graph isomorphism as clique search

Let $G = (V, E)$ be an undirected graph, where V is the set of vertices and $E \subseteq V \times V$ is the set of edges. The *order* of G is the number of its vertices, and its *size* is the number of edges. Two vertices $i, j \in V$ are said to be *adjacent* if $(i, j) \in E$. The *adjacency matrix* of G is the $n \times n$ symmetric matrix $A = (a_{ij})$ defined as follows:

$$a_{ij} = \begin{cases} 1, & \text{if } (i,j) \in E \ , \\ 0, & \text{otherwise} \ . \end{cases}$$

[1]It should be pointed out, however, that these are worst-case results, and there are certain classes of graphs for which the problem is solvable in polynomial time (Grötschel, Lovász and Schrijver, 1988; Pardalos and Xue, 1994).

The *degree* of a vertex $i \in V$, denoted by $\deg(i)$, is the number of vertices adjacent to it, i.e., $\deg(i) = \sum_j a_{ij}$.

Given two graphs $G' = (V', E')$ and $G'' = (V'', E'')$, an *isomorphism* between them is any bijection $\phi : V' \to V''$ such that $(i, j) \in E' \Leftrightarrow (\phi(i), \phi(j)) \in E''$, for all $i, j \in V'$. Two graphs are said to be *isomorphic* if there exists an isomorphism between them. The graph isomorphism problem is therefore to decide whether two graphs are isomorphic and, in the affirmative, to find an isomorphism. The maximum common subgraph problem is more general and difficult (Garey and Johnson, 1979), and includes the graph isomorphism problem as a special case. It consists of finding the largest isomorphic subgraphs of G' and G''. A simpler version of this problem is to find a maximal common subgraph, i.e., an isomorphism between subgraphs which is not included in any larger subgraph isomorphism.

Barrow and Burstall (1976), and also Kozen (1978), introduced the notion of an *association graph* as a useful auxiliary graph structure for solving general graph/subgraph isomorphism problems.

Definition 1 *The association graph derived from graphs $G' = (V', E')$ and $G'' = (V'', E'')$ is the undirected graph $G = (V, E)$ defined as follows:*

$$V = V' \times V''$$

and

$$E = \left\{ ((i,h),(j,k)) \in V \times V \ : \ i \neq j, \ h \neq k, \ \textit{and} \ (i,j) \in E' \Leftrightarrow (h,k) \in E'' \right\} .$$

Given an arbitrary undirected graph $G = (V, E)$, a subset of vertices C is called a *clique* if all its vertices are mutually adjacent, i.e., for all $i, j \in C$ we have $(i, j) \in E$. A clique is said to be *maximal* if it is not contained in any larger clique, and *maximum* if it is the largest clique in the graph. The *clique number*, denoted by $\omega(G)$, is defined as the cardinality of the maximum clique.

The following result establishes an equivalence between the graph isomorphism problem and the maximum clique problem.

Theorem 2 *Let $G' = (V', E')$ and $G'' = (V'', E'')$ be two graphs of order n, and let G be the corresponding association graph. Then, G' and G'' are isomorphic if and only if $\omega(G) = n$. In this case, any maximum clique of G induces an isomorphism between G' and G'', and vice* *versa. In general, maximal/maximum cliques in G are in one-to-one correspondence with maximal/maximum common subgraph isomorphisms between G' and G'', respectively.*

Proof: Suppose that the two graphs are isomorphic and let ϕ be an isomorphism between them. Then the subset of vertices of G defined as $C_\phi = \{(i, \phi(i)) : \forall i \in V'\}$ is clearly a maximum clique of cardinality n. Conversely, let C be an n-vertex maximum clique of G and for each $(i, h) \in C$ define $\phi(i) = h$. Then, because of the way the association graph is constructed, it is clear that ϕ is an isomorphism between G' and G''. The proof for the general case is analogous. □

2.2 Continuous formulation of the maximum clique problem

Let $G = (V, E)$ be an arbitrary undirected graph of order n, and let S_n denote the standard simplex of $\mathbb{R}^n$:

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

Given a subset of vertices C of G we shall denote by $\mathbf{x}^c$ its *characteristic vector*, which is the point in S_n defined as

$$x_i^c = \begin{cases} 1/|C|, & \text{if } i \in C \\ 0, & \text{otherwise} \end{cases}$$

where $|C|$ denotes the cardinality of C.

Now, consider the following quadratic function

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{x}^T A \mathbf{x} \\ &= \textstyle\sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j \ . \end{aligned} \tag{1}$$

where $A = (a_{ij})$ is the adjacency matrix of G, and "T" denotes transposition. A point $\mathbf{x}^* \in S_n$ is said to be a *global* maximizer of f in S_n if $f(\mathbf{x}^*) \geq f(\mathbf{x})$, for all $\mathbf{x} \in S_n$. It is said to be a *local* maximizer if there exists an $\varepsilon > 0$ such that $f(\mathbf{x}^*) \geq f(\mathbf{x})$ for all $\mathbf{x} \in S_n$ whose distance from $\mathbf{x}^*$ is less than ε, and if $f(\mathbf{x}^*) = f(\mathbf{x})$ implies $\mathbf{x}^* = \mathbf{x}$, then $\mathbf{x}^*$ is said to be a *strict* local maximizer.

The Motzkin-Straus theorem (Motzkin and Straus, 1965) establishes a remarkable connection between global (local) maximizers of the function f in S_n and maximum (maximal) cliques of G. Specifically, it states that a subset of vertices C of a graph G is a maximum clique if and only if its characteristic vector $\mathbf{x}^c$ is a global maximizer of f on S_n. A similar relationship holds between (strict) local maximizers and maximal cliques (Gibbons et al., 1997; Pelillo and Jagota, 1995). This result has an intriguing computational significance in that it allows us to shift from the discrete to the continuous domain in an elegant manner. Such a reformulation is attractive for several reasons. It not only allows us to exploit the full arsenal of continuous optimization techniques, thereby leading to the development of new algorithms, but may also reveal unexpected theoretical properties. Additionally, continuous optimization methods are often described in terms of (ordinary) differential equations, and are therefore potentially implementable in analog circuitry. The Motzkin-Straus theorem has served as the basis of many clique-finding procedures (Bomze, Pelillo and Giacomini, 1997; Bomze et al., 1998; Gibbons, Hearn and Pardalos, 1996; Pardalos and Phillips, 1990; Pelillo, 1995), and has also been used to determine theoretical bounds on the clique number (Pardalos and Phillips, 1990; Wilf, 1986).

One drawback associated with the original Motzkin-Straus formulation relates to the existence of spurious solutions, i.e., maximizers of f which are not in the form of characteristic vectors. This was observed empirically by Pardalos and Phillips (1990) and has more recently been formalized by Pelillo and Jagota (1995). In principle, spurious solutions represent a problem since, while providing information about the *cardinality* of the maximum clique, do not allow us to easily extract its vertices. Fortunately, there is straightforward solution to this problem which has recently been introduced and studied by Bomze (1997). Consider the following regularized version of function f:

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_i x_j + \frac{1}{2}\sum_{i=1}^{n} x_i^2 \tag{2}$$

which is obtained from (1) by substituting the adjacency matrix A of G with

$$\hat{A} = A + \frac{1}{2}I_n$$

where I_n is the $n \times n$ identity matrix. The following is the spurious-free counterpart of the original Motzkin-Straus theorem (see (Bomze, 1997) for proof).

Theorem 3 *Let C be a subset of vertices of a graph G, and let $\mathbf{x}^c$ be its characteristic vector. Then the following statements hold:*

(a) C is a maximum clique of G if and only if $\mathbf{x}^c$ is a global maximizer of the function $\hat{f}$ over the simplex S_n. In this case, $\omega(G) = 1/2(1 - \hat{f}(\mathbf{x}^c))$.

(b) C is a maximal clique of G if and only if $\mathbf{x}^c$ is a local maximizer of $\hat{f}$ in S_n.

(c) All local (and hence global) maximizers of $\hat{f}$ over S_n are strict.

Unlike the Motzkin-Straus formulation, the previous result guarantees us that *all* maximizers of $\hat{f}$ on S_n are strict, and are characteristic vectors of maximal/maximum cliques in the graph. In an exact sense, therefore, a one-to-one correspondence exists between maximal cliques and local maximizers of $\hat{f}$ in S_n on the one hand, and maximum cliques and global maximizers on the other hand. This solves the spurious solution problem in a definitive manner.

2.3 A quadratic program for graph isomorphism

In light of the above discussion, it is now a straightforward exercise to formulate the graph isomorphism problem in terms of a standard quadratic programming problem. Let G' and G'' be two arbitrary graphs of order n, and let A denote the adjacency matrix of the corresponding association graph, whose order is $N = n^2$. The graph isomorphism problem is equivalent to the following program:

$$\begin{array}{ll} \text{maximize} & \hat{f}(\mathbf{x}) = \mathbf{x}^T(A + \frac{1}{2}I_N)\mathbf{x} \\ \text{subject to} & \mathbf{x} \in S_N \end{array} \tag{3}$$

More precisely, the following result holds, which is a straightforward consequence of Theorems 2 and 3.

Theorem 4 *Let $G' = (V', E')$ and $G'' = (V'', E'')$ be two graphs of order n, and let $\mathbf{x}^*$ be a global solution of program (3), where A is the adjacency matrix of the association graph of G' and G''. Then, G' and G'' are isomorphic if and only if $\hat{f}(\mathbf{x}^*) = 1 - 1/2n$. In this case, any global solution to (3) induces an isomorphism between G' and G'', and vice versa. In general, local/global solutions to (3) are in one-to-one correspondence with maximal/maximum common subgraph isomorphisms between G' and G'', respectively.*

Note that the adjacency matrix $A = (a_{ih,jk})$ of the association graph can be explicitly written as follows:

$$a_{ih,jk} = \begin{cases} 1 - (a'_{ij} - a''_{hk})^2, & \text{if } i \neq j \text{ and } h \neq k \\ 0, & \text{otherwise} \end{cases}$$

where $A' = (a'_{ij})$ and $A'' = (a''_{hk})$ are the adjacency matrices of G' and G'', respectively. The regularized Motzkin-Straus objective function $\hat{f}$ therefore becomes:

$$\hat{f}(\mathbf{x}) = \sum_{i,h} \sum_{j \neq i} \sum_{k \neq h} a'_{ij} a''_{hk} x_{ih} x_{jk} + \sum_{i,h} \sum_{j \neq i} \sum_{k \neq h} (1 - a'_{ij})(1 - a''_{hk}) x_{ih} x_{jk} + \frac{1}{2} \sum_{i,h} x_{ih}^2 \ . \qquad (4)$$

Many interesting observations about the previous objective function can be made. It consists of three terms. The first is identical to the one used in (Mjolsness, Gindi and Anandan, 1989; Gold and Rangarajan, 1996; Rangarajan, Gold and Mjolsness, 1996; Rangarajan and Mjolsness, 1996), which derives from the so-called *rectangle rule*. Intuitively, by restricting ourselves to binary variables $x_{ih} \in \{0, 1\}$, it simply counts the number of consistent "rectangles" between G' and G'' that are induced by the tentative solution $\mathbf{x}$. The second term is new and, by analogy with the rectangle rule, can be derived from what can be called the *anti-rectangle* rule: in case of binary variables, it counts the number of rectangles between the complements of the original graphs.[2] Finally, the third term in (4), which has been added to avoid spurious solutions in the Motzkin-Straus program, is just the self-amplification term introduced in a different context by Yuille and Kosowsky (1994) for a related purpose, i.e., ensuring that the minimizers of a generic quadratic function in the unit hypercube lie at the vertices. The self-amplification term has also been employed recently in (Rangarajan, Gold and Mjolsness, 1996; Rangarajan and Mjolsness, 1996). Like ours, the self-amplification term has the form $\gamma \sum_{i,h} x_{ih}^2$, but the parameter γ depends on the structure of the quadratic program matrix. In our case $\gamma = \frac{1}{2}$, and it can easily be proved that Theorem 3 holds true for all $\gamma \in (0, 1)$. Its choice is therefore independent of the structure of the matrix A, and only affects the basins of attraction around local optima.[3]

3 Replicator equations and graph isomorphism

3.1 The model and its properties

Replicator equations have been developed and studied in the context of evolutionary game theory, a discipline pioneered by J. Maynard Smith (1982) which aims to model the evolution of animal behavior using the principles and tools of game theory. In this section we discuss the basic intuition behind replicator equations and present a few theoretical properties that

[2]The complement of a graph $G = (V, E)$ is the graph $\overline{G} = (V, \overline{E})$ such that $(i, j) \in \overline{E} \Leftrightarrow (i, j) \notin E$.

[3]The effects of allowing γ to take on negative values, and of varying it during the optimization process are studied in (Bomze et al., 1998).

will be instrumental in the subsequent development of our graph isomorphism algorithm. For a more systematic treatment see (Hofbauer and Sigmund, 1988; Weibull, 1995).

Consider a large population of individuals belonging to the same species which compete for a particular limited resource, such as food, territory, etc. This kind of conflict is modeled as a game, the players being pairs of randomly selected population members. In contrast to traditional application fields of game theory, such as economics or sociology (Luce and Raiffa, 1957), players here do not behave "rationally," but act instead according to a "pre-programmed" behavior pattern, or *pure strategy*. Reproduction is assumed to be asexual, which means that, apart from mutation, offsprings will inherit the same genetic material, and hence behavioral phenotype, as its parent. Let $J = \{1, \cdots, n\}$ be the set of pure strategies and, for all $i \in J$, let $x_i(t)$ be the relative frequency of population members playing strategy i, at time t. The *state* of the system at time t is simply the vector $\mathbf{x}(t) = (x_1(t), \cdots, x_n(t))^T$.

One advantage of applying game theory to biology is that the notion of "utility" is much simpler and clearer than in human contexts. Here, a player's utility can simply be measured in terms of Darwinian fitness or reproductive success, i.e., the player's expected number of offsprings. Let $W = (w_{ij})$ be the $n \times n$ "payoff" (or fitness) matrix. Specifically, for each pair of strategies $i, j \in J$, w_{ij} represents the payoff of an individual playing strategy i against an opponent playing strategy j. Without loss of generality, we shall assume that the payoff matrix is nonnegative, i.e., $w_{ij} \geq 0$ for all $i, j \in J$. At time t, the average payoff of strategy i is given by:

$$\pi_i(t) = \sum_{j=1}^{n} w_{ij} x_j(t)$$

while the mean payoff over the entire population is $\sum_{i=1}^{n} x_i(t)\pi_i(t)$.

In evolutionary game theory the assumption is made that the game is played over and over, generation after generation, and that the action of natural selection will result in the evolution of the fittest strategies. If successive generations blend into each other, the evolution of behavioral phenotypes can be described by the following set of differential equations (Taylor and Jonker, 1978):

$$\dot{x}_i(t) = x_i(t) \left(\pi_i(t) - \sum_{j=1}^{n} x_j(t)\pi_j(t) \right) \quad , \quad i = 1 \ldots n \tag{5}$$

where a dot signifies derivative with respect to time. The basic idea behind this model is that the average rate of increase $\dot{x}_i(t)/x_i(t)$ equals the difference between the average fitness of strategy i and the mean fitness over the entire population. It is straightforward to show that the simplex S_n is invariant under equation (5) or, in other words, any trajectory starting in S_n will remain in S_n. To see this, simply note that $\frac{d}{dt}\sum_i x_i(t) = \sum_i \dot{x}_i(t) = 0$, which means that the interior of S_n (i.e., the set defined by $x_i > 0$, for all $i = 1 \ldots n$) is invariant. The additional observation that the boundary too is invariant, completes the proof.

Similar arguments provide a rationale for the following discrete-time version of the replicator dynamics, assuming non-overlapping generations:

$$x_i(t+1) = \frac{x_i(t)\pi_i(t)}{\sum_{j=1}^{n} x_j(t)\pi_j(t)} \ , \quad i = 1 \ldots n \ . \tag{6}$$

Because of the non-negativity of the fitness matrix W and the normalization factor, this system too makes the simplex S_n invariant as its continuous counterpart.

A point $\mathbf{x} = \mathbf{x}(t)$ is said to be a *stationary* (or equilibrium) point for our dynamical systems, if $\dot{x}_i(t) = 0$ in the continuous-time case, and $x_i(t+1) = x_i(t)$ in the discrete-time case ($i = 1 \ldots n$). Moreover, a stationary point is said to be *asymptotically stable* if any trajectory starting in its vicinity will converge to it as $t \to \infty$. It turns out that both the continuous-time and discrete-time replicator dynamics have the same set of stationary points, namely all the points in S_n satisfying the condition

$$x_i(t)\left(\pi_i(t) - \sum_{j=1}^{n} x_j(t)\pi_j(t)\right) = 0 \ , \quad i = 1 \ldots n \tag{7}$$

or, equivalently, $\pi_i(t) = \sum_{j=1}^{n} x_j(t)\pi_j(t)$ whenever $x_i > 0$.

Equations (5) and (6) arise independently in different branches of theoretical biology (Hofbauer and Sigmund, 1988). In population ecology, for example, the famous Lotka-Volterra equations for predator-prey systems turn out to be equivalent to the continuous-time dynamics (5), under a simple barycentric transformation and a change in velocity. In population genetics they are known as *selection equations* (Crow and Kimura, 1970). In this case, each x_i represents the frequency of the i-th allele A_i and the payoff w_{ij} is the "fitness" of genotype A_iA_j. Here the fitness matrix W is always symmetric. The discrete-time dynamical equations turn out to be a special case of a general class of dynamical systems introduced by Baum and Eagon (1967) and studied by Baum and Sell (1968) in the context of Markov chain theory. They also represent an instance of the so-called *relaxation labeling processes*, a class of parallel, distributed algorithms developed in computer vision to solve (continuous) constraint satisfaction problems (Rosenfeld, Hummel and Zucker, 1976; Hummel and Zucker, 1983). An independent connection between dynamical systems such as relaxation labeling and Hopfieldstyle networks and game theory has recently been described by Miller and Zucker (1991, 1992).

We are now interested in studying the dynamical properties of replicator systems; it is these properties that will allow us to employ them for solving the graph isomorphism problem. The following theorem states that under replicator dynamics the population's average fitness always increases, provided that the payoff matrix is symmetric (in game theory terminology, this situation is referred to as a doubly symmetric game).

Theorem 5 *Suppose that the (nonnegative) payoff matrix W is symmetric (i.e., $w_{ij} = w_{ji}$ for all $i, j = 1 \ldots n$). The quadratic polynomial F defined as*

$$F(\mathbf{x}) = \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} x_i x_j \tag{8}$$

is strictly increasing along any non-constant trajectory of both continuous-time (5) and discrete-time (6) replicator equations. In other words, for all $t \geq 0$ we have

$$\frac{d}{dt} F(\mathbf{x}(t)) > 0$$

for system (5), and

$$F(\mathbf{x}(t+1)) > F(\mathbf{x}(t))$$

for system (6), unless $\mathbf{x}(t)$ is a stationary point. Furthermore, any such trajectory converges to a (unique) stationary point.

The previous result is known in mathematical biology as the Fundamental Theorem of Natural Selection (Crow and Kimura, 1970; Hofbauer and Sigmund, 1988; Weibull, 1995) and, in its original form, traces back to Fisher (1930). As far as the discrete-time model is concerned, it can be regarded as a straightforward implication of the Baum-Eagon theorem (Baum and Eagon, 1967; Baum and Sell, 1968) which is valid for general polynomial functions over product of simplices. Waugh and Westervelt (1993) also proved a similar result for a related class of continuous- and discrete-time dynamical systems. In the discrete-time case, however, they put bounds on the eigenvalues of W in order to achieve convergence to fixed points.

The fact that all trajectories of the replicator dynamics converge to a stationary point has been proved more recently (Losert and Akin, 1983; Lyubich, Maistrowskii and Ol'khovskii, 1980). However, in general, not all stationary points are local maximizers of F on S_n. The vertices of S_n, for example, are all stationary points for (5) and (6) whatever the landscape of

F. Moreover, there may exist trajectories which, starting from the interior of S_n, eventually approach a saddle point of F. However, a result recently proved by Bomze (1997) asserts that all asymptotically stable stationary points of replicator dynamics correspond to (strict) local maximizers of $\hat{f}$ on S_n, and vice versa.

3.2 Application to graph isomorphism problems

The properties discussed in the preceding subsection naturally suggest using replicator equations as a useful heuristic for the graph isomorphism problem. Let $G' = (V', E')$ and $G'' = (V'', E'')$ be two graphs of order n, and let A denote the adjacency matrix of the corresponding N-vertex association graph G. By letting

$$W = A + \frac{1}{2} I_N$$

we know that the replicator dynamical systems, starting from an arbitrary initial state, will iteratively maximize the function $\hat{f}(\mathbf{x}) = \mathbf{x}^T (A + \frac{1}{2} I_N)\mathbf{x}$ in S_N, and will eventually converge to a strict local maximizer which, by virtue of Theorem 3 will then correspond to the characteristic vector of a maximal clique in the association graph.[4] We know from Theorem 4 that this will in turn induce an isomorphism between two subgraphs of G' and G'' which is maximal, in the sense that there is no other isomorphism between subgraphs of G' and G'' which includes the one found. Clearly, in theory there is no guarantee that the converged solution will be a *global* maximizer of $\hat{f}$, and therefore that it will induce a maximum isomorphism between the two original graphs. However, previous experimental work done on the maximum clique problem (Bomze, Pelillo and Giacomini, 1997; Pelillo, 1995), and also the results presented in this paper, suggest that the basins of attraction of global maximizers are quite large, and very frequently the algorithm converges to one of them. Without any heuristic information about the optimal solution, it is customary to start out the replicator process from the barycenter of the simplex, i.e., the vector $(\frac{1}{N}, \cdots, \frac{1}{N})^T$. This choice ensures that no particular solution is favored.

It is interesting to note that the emergent matching strategy of our replicator model is identical to the one adopted by Simić's algorithm (Simić, 1991), which is speculated to be similar to that employed by humans in solving matching problems. Specifically, it seems

[4]Because of the presence of saddle points, the algorithm may occasionally converge toward one such points. However, since the set of saddle points is of measure zero, this happens with probability tending to zero.

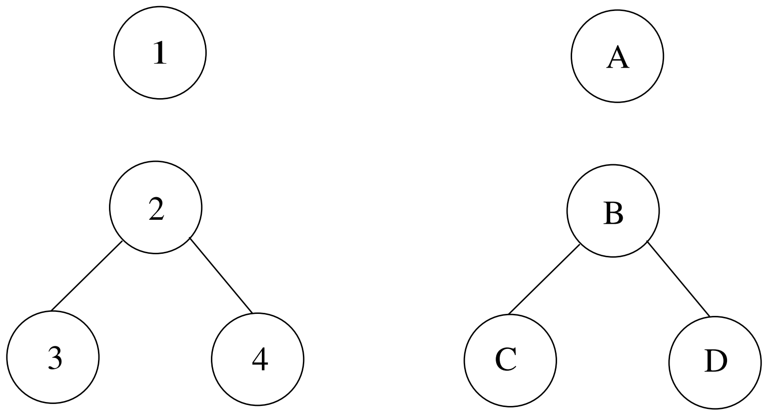


Figure 1: A pair of isomorphic graphs.

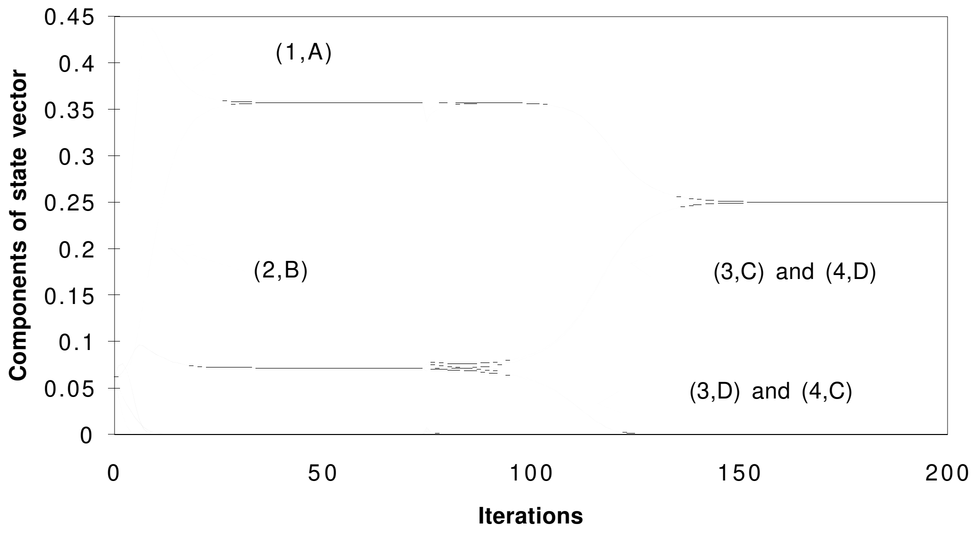


Figure 2: Evolution of the components of the state vector $\mathbf{x}(t)$ for the graphs in Fig. 1, using the replicator dynamics (6).

that the algorithm first tries to match what Simić called the *notable* vertices, that is those vertices having highest or lowest connectivity. To illustrate, consider two vertices $i \in V'$ and $h \in V''$, and assume for simplicity that they have the same degree, $\deg(i) = \deg(h)$. It is easy to show that the corresponding vertex in the association graph has at most degree $\deg(i,h) = \binom{\deg(i)}{2} + \binom{n-1-\deg(i)}{2} = \deg^2(i) - (n-1)\deg(i) + \binom{n-1}{2}$, which attains its minimum value when $\deg(i) = \frac{n-1}{2}$, and maximum value when $\deg(i)$ equals 0 or $n-1$. It follows therefore that pairs of notable vertices give rise to vertices in the association graph having largest degree. Now, consider what happens at the very first iterations of our clique-finding relaxation process assuming, as is customary, that it is started from the barycenter of S_N. At $t = 0$, the average payoff of a vertex (i,h) in the association graph is $\pi_{ih}(0) = \frac{1}{N}\sum_{jk} a_{ih,jk} + \frac{1}{2N} = \frac{1}{2N}(2\deg(i,h) + 1)$. Because of the payoff monotonicity property of replicator dynamics (*cfr.* equations (10) and (12) in the next section) this implies that at the very beginning of the relaxation process the components corresponding to pairs of notable vertices will grow at a higher rate, thereby imposing a sort of partial ordering over the set of possible assignments. Clearly this simplified picture is no longer valid after the first few iterations, when local information begins to propagate.

We illustrate this behavior with the aid of a simple example. Consider the two isomorphic graphs in Fig. 1. Our matching strategy would suggest to first match vertex 1 to vertex A, then 2 to B, and finally either 3 to C and 4 to D, or 3 to D and 4 to C. These are the only possible isomorphisms between the two graphs. As shown in Fig. 2, this is exactly what our algorithm accomplishes. The figure plots the evolution of each component of the state vector $\mathbf{x}(t)$, a 16-dimensional vector, under the replicator dynamics (6). Observe how, after rapidly trying to match 1 to A and 2 to B, it converges to a saddle point, which indeed incorporates the information regarding the two possible isomorphisms. After a slight perturbation, at around the 75th step, the process makes a choice and quickly converges to one of the two correct solutions.

3.3 Experimental results

In order to assess the effectiveness of the proposed approach, extensive simulations were performed over randomly generated graphs of various connectivities. Random graphs represent a useful benchmark not only because they are not constrained to any particular application, but also because it is simple to replicate experiments and hence to make comparisons with other algorithms. Before going into the details of the experiments, however, we need to enter a preliminary caveat.

It is often said that random graph isomorphism is trivial. Essentially, this claim is based on a result due to Babai, Erdös and Selkow (1980) which shows that a straightforward, linear-time graph isomorphism algorithm does work for *almost all* random graphs.[5] It should be pointed out, however, that there are various probability models for random graphs (Palmer, 1985). The one adopted by Babai considers random graphs as uniformly distributed random variables, i.e., they assume that the probability of generating any n-vertex graph equals $2^{-\binom{n}{2}}$. By contrast, the customary way in which random graphs are generated leads to a distribution which only in a special case is uniform. Specifically, given a parameter p $(0 < p < 1)$ which represents the expected connectivity, a graph of order n is generated by randomly entering edges between the vertices with probability p. Note that p is related to the expected size of the resulting graph, which indeed is $\binom{n}{2}p$. It is straightforward to see that in so doing the probability that a graph of order n and size s be generated is given by $p^s(1-p)^{\binom{n}{2}-s}$, which only in the case $p = \frac{1}{2}$ equals Babai's uniform distribution. Now, the results presented in (Babai, Erdös and Selkow, 1980) are based on the observation that, using a uniform probability model, the degrees of the vertices have large variability and this is in fact the key to their algorithm. In the non-uniform probability model, the degree random variable has variance $(n-1)p(1-p)$ and it is no accident that attains its largest value exactly at $p = \frac{1}{2}$. However, as p moves away from $\frac{1}{2}$ the variance becomes smaller and smaller, tending to 0 as p approaches 0 or 1. As a result, Babai's arguments are no longer applicable. It seems therefore that, using the customary graph generation model, random graph isomorphism is not as trivial as is generally believed, especially for very sparse and very dense graphs. As a matter of fact, the experience reported in (Rangarajan, Gold and Mjolsness, 1996; Rangarajan and Mjolsness, 1996; Simić, 1991) and also the results presented below, provide support to this claim.

In the experiments reported here, the algorithm was started from the barycenter of the simplex and stopped when either a maximal clique (i.e., a local maximizer of $\hat{f}$ on S_n) was found or the distance between two successive points was smaller than a fixed threshold, which was set to 10^{-17}. 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

[5]A property is said to hold for *almost all* graphs, if the probability that the property holds tends to 1 as the order of the graph approaches infinity.

between local maximizers and maximal cliques, this situation corresponds to convergence to a saddle point. All the experiments were run on a Sparc20.

Undirected 100-vertex random graphs were generated with expected connectivities ranging from 1% to 99%. Specifically, the values of the edge-probability p were as follows: 0.01, 0.03, 0.05, 0.95, 0.97, 0.99, and from 0.1 to 0.9 in steps of 0.1. For each connectivity value, 100 graphs were produced and each of them had its vertices randomly permuted so as to obtain a pair of isomorphic graphs. Overall, therefore, 1500 pairs of isomorphic graphs were generated.

To keep the order of the association graph as low as possible, its vertex set was constructed as follows:

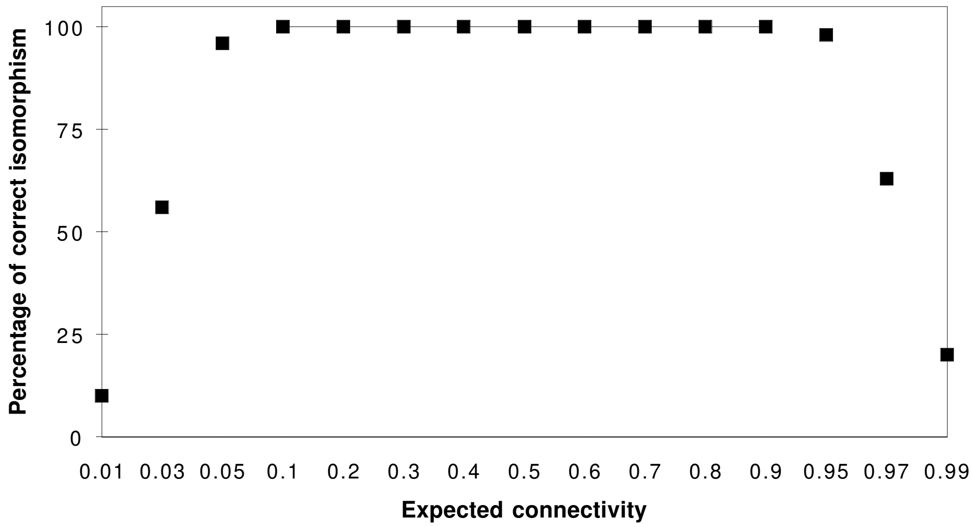
$$V = \{(i,h) \in V' \times V'' \ : \ \deg(i) = \deg(h)\} \ ,$$

the edge set E being defined as in Definition 1. It is straightforward to see that when the graphs are isomorphic Theorem 2 continues to hold, since isomorphisms preserve the degree property of the vertices. This simple heuristic may significantly reduce the dimensionality of the search space.

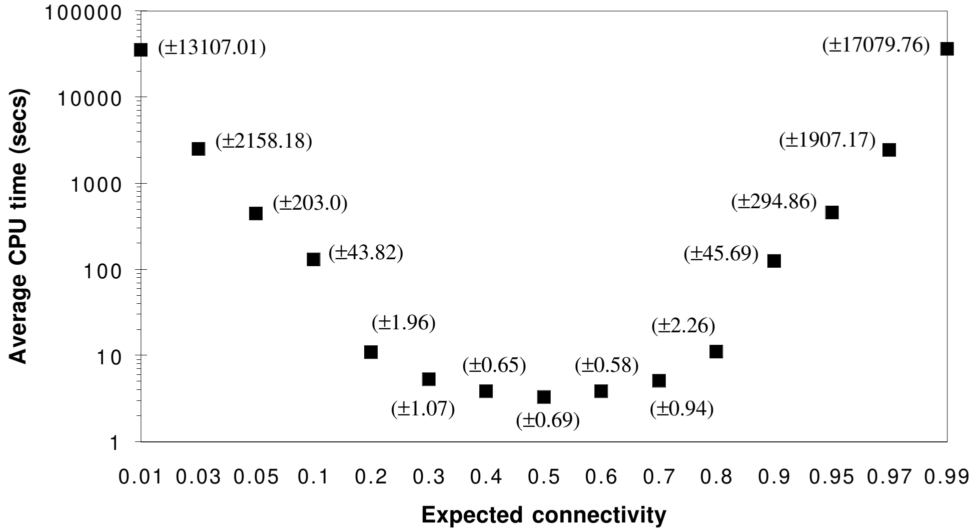
Each pair of isomorphic graphs was given as input to the replicator model and, after convergence, a success was recorded when the cardinality of the returned clique was equal to the order of the graphs given as input, i.e., 100.[6] Because of the stopping criterion employed, this guarantees that a maximum clique, and therefore a correct isomorphism, was found. Fig. 3(a) plots the proportion of successes as a function of p, while Fig. 3(b) shows the average CPU time (in logarithmic scale) taken by the algorithm to converge.

These results are significantly superior to those reported by Simić (1991) who obtained poor results at connectivities less than 40% even on smaller graphs (i.e., up to 75 vertices). They also compare favorably with the results obtained more recently by Rangarajan et al. (1996) on 100-vertex random graphs for connectivities up to 50%. Specifically, at 1% and 3% connectivities they report a percentage of correct isomorphisms of about 0% and 30%, respectively. Using our approach we obtained, on the same kind of graphs, a percentage of success of 10% and 56%, respectively. Rangarajan and Mjolsness (1996) also run experiments on 100-vertex random graphs with various connectivities, using a powerful Lagrangian relaxation network. Except for a few instances, they always obtained a correct solution. The computational time required by their model, however, turns out to largely exceed ours. As an example, the average time

[6]Due to the high computational time required, in the $p = 0.01$ and $p = 0.99$ cases the algorithm was tested only on 10 pairs, instead of 100.



(a)



(b)

Figure 3: Results obtained over 100-vertex graphs of various connectivities, using dynamics (6). (a) Percentage of correct isomorphisms. (b) Average computational time taken by the replicator equations. The vertical axis is in logarithmic scale, and the numbers in parentheses represent the standard deviation.

taken by their algorithm to match two 100-vertex 50%-connectivity graphs was about 30 minutes on an SGI workstation. As shown in Fig. 3(b), we obtained identical results in about 3 seconds. However, for very sparse and very dense graphs our algorithm becomes extremely slow. In the next section, we present an exponential version of our replicator dynamics which turns out to be dramatically faster and even more accurate than the classical model (6).

It should be emphasized that all the algorithms mentioned above do incorporate sophisticated annealing mechanisms to escape from poor local minima. By contrast, in the presented work no attempt was made to prevent the algorithm from converging to such solutions. It seems that, as far as the graph isomorphism problem is concerned, global maximizers of the Motzkin-Straus objective have large basins of attraction. A similar observation was also made in connection to earlier experiments concerning the maximum clique problem (Bomze, Pelillo and Giacomini, 1997; Pelillo, 1995).

4 Faster replicator dynamics

Recently, there has been much interest in evolutionary game theory around the following exponential version of replicator equations, which arises as a model of evolution guided by imitation (Hofbauer, 1995; Hofbauer and Weibull, 1996; Weibull, 1994; Weibull, 1995):

$$\dot{x}_i(t) = x_i(t)\left(\frac{e^{\kappa\pi_i(t)}}{\sum_{j=1}^{n} x_j(t)e^{\kappa\pi_j(t)}} - 1\right) \quad , \quad i = 1 \dots n \tag{9}$$

where κ is a positive constant. As κ tends to 0, the orbits of this dynamics approach those of the standard, "first-order" replicator model (5), slowed down by the factor κ; moreover, for large values of κ the model approximates the so-called "best-reply" dynamics (Hofbauer and Weibull, 1996). It is readily seen that dynamics (9) is *payoff monotonic* (Weibull, 1995), which means that

$$\frac{\dot{x}_i(t)}{x_i(t)} > \frac{\dot{x}_j(t)}{x_j(t)} \Leftrightarrow \pi_i(t) > \pi_j(t) \tag{10}$$

for $i, j = 1 \dots n$. This amounts to stating that during the evolution process the components corresponding to higher payoffs will increase at higher rate. Observe that the first-order replicator model (5) also is payoff monotonic. The class of payoff monotonic dynamics possesses several interesting properties (Weibull, 1995). In particular, they all have the same set of stationary points, which are characterized by (7). Moreover, when the fitness matrix W is symmetric, the average population payoff defined in (8) is also strictly increasing, as in the

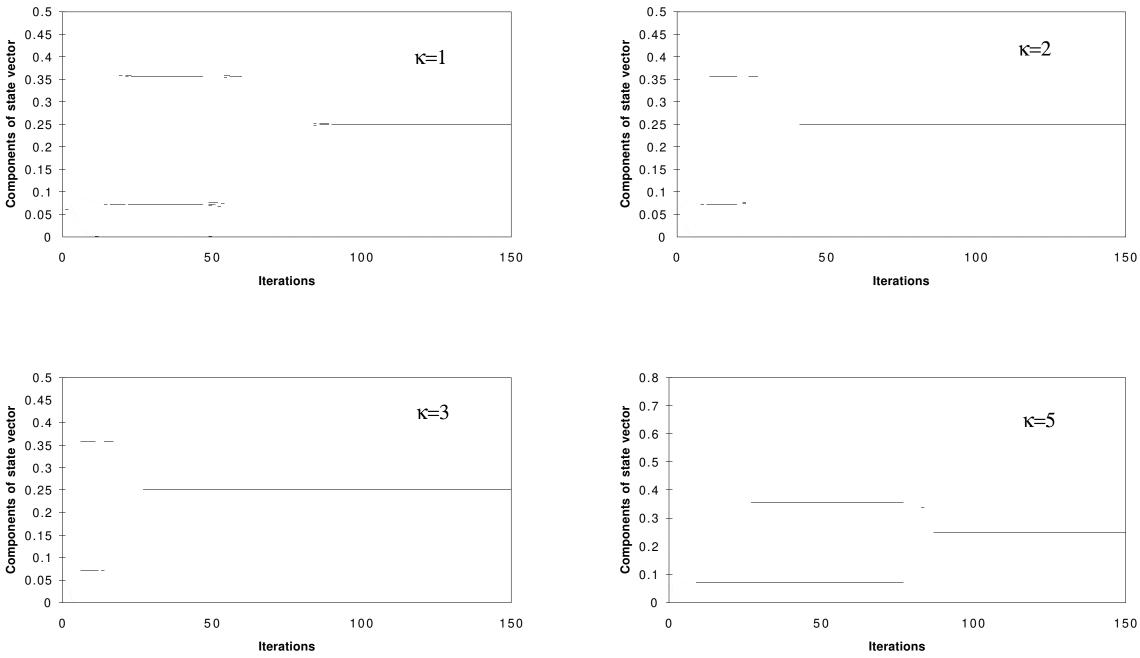


Figure 4: Evolution of the components of the state vector $\mathbf{x}(t)$ for the graphs in Fig. 1, using the exponential replicator model (11), for different values of the parameter κ.

first-order case (see (Hofbauer, 1995) for proof). After discussing various properties of payoff monotonic dynamics, Hofbauer (1995) has recently concluded that they behave essentially in the same way as the standard replicator equations, the only difference being the size of the basins of attraction around stable equilibria.

A customary way of discretizing equation (9) is given by the following difference equations (Cabrales and Sobel, 1992; Gaunersdorfer and Hofbauer, 1995), which is also similar to the "self-annealing" dynamics recently introduced by Rangarajan (1997):

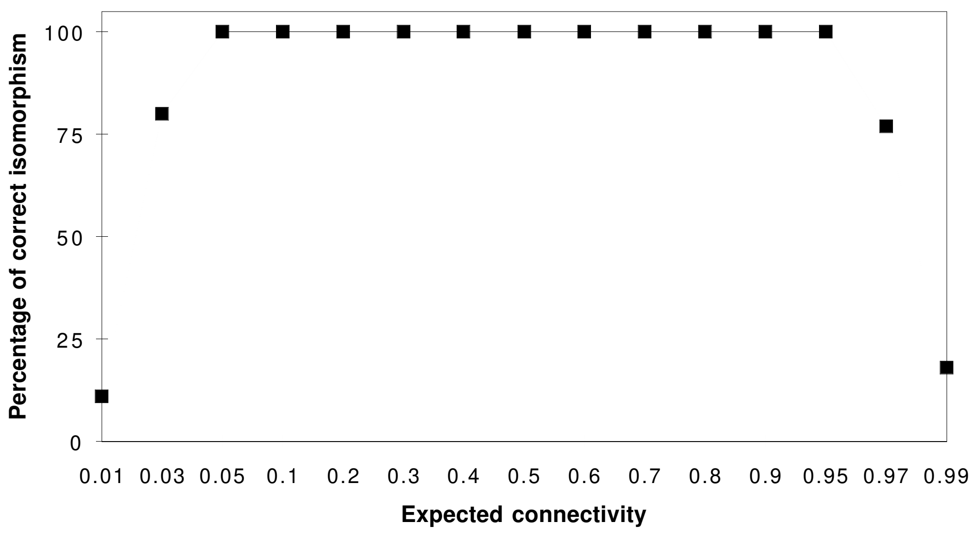
$$x_i(t+1) = \frac{x_i(t)e^{\kappa\pi_i(t)}}{\sum_{j=1}^{n} x_j(t)e^{\kappa\pi_j(t)}}\ , \quad i = 1 \ldots n\ . \tag{11}$$

As its continuous counterpart, this dynamics is payoff monotonic, i.e.,

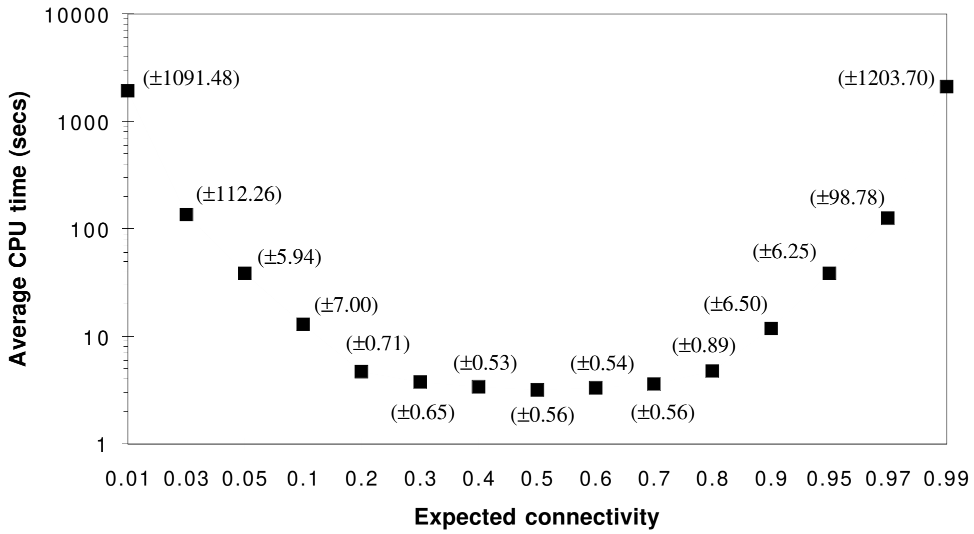
$$\frac{x_i(t+1) - x_i(t)}{x_i(t)} > \frac{x_j(t+1) - x_j(t)}{x_j(t)} \ \Leftrightarrow\ \pi_i(t) > \pi_j(t)\ , \tag{12}$$

for all $i, j = 1 \ldots n$. Observe that the standard discrete-time equations (6) also possess this property.

¿From our computational perspective, exponential replicator dynamics are particularly attractive because, as demonstrated by the extensive numerical results reported below, they seem to be considerably faster and even more accurate than the standard, first-order model.



(a)



(b)

Figure 5: Results obtained over 100-vertex graphs of various connectivities, using the exponential dynamics (11). (a) Percentage of correct isomorphisms. (b) Average computational time taken by the replicator equations. The vertical axis is in logarithmic scale, and the numbers in parentheses represent the standard deviation.

To illustrate, in Fig. 4 the behavior of the dynamics (11) in matching the simple graphs of Fig. 1 is shown, for various choices of the parameter κ. Notice how the qualitative behavior of the algorithm is the same as the first-order model, but now convergence is dramatically faster (*cfr.* Fig. 2). In this example, the process becomes unstable when $\kappa = 5$, suggesting, as expected, that the choice of this parameter is a trade-off between speed and stability. Unfortunately, there is no theoretical principle to properly choose this parameter.

To practically test the validity of this new model on a larger scale, we conducted a second series of experiments over the same 1500 graphs generated for testing the first-order dynamics. The discrete-time equations (11) were used, and the parameter κ was heuristically set to 10. The process was started from the barycenter of the simplex and stopped using the same criterion used in the previous set of experiments. Fig. 5 shows the percentage of successes obtained for the various connectivity values, and the average CPU time taken by the algorithm to converge (in logarithmic scale). It is evident from these results that the exponential replicator system (11) may be dramatically faster than the first-order model (6), and may also provide better results.

5 Conclusions

In this paper, we have developed a new energy-minimization framework for the graph isomorphism problem which is centered around an equivalent maximum clique formulation and the Motzkin-Straus theorem, a remarkable result which establishes an elegant connection between the maximum clique problem and a certain standard quadratic program. The attractive feature of the proposed formulation is that a clear one-to-one correspondence exists between the solutions of the quadratic program and those in the original, discrete problem. We have then introduced the so-called "replicator" equations, a class of continuous- and discrete-time dynamical systems developed in evolutionary game theory and various other branches of theoretical biology, and have shown how they naturally lend themselves to approximately solving the Motzkin-Straus program. The extensive experimental results presented show that, despite their simplicity and their inherent inability to escape from local optima, replicator dynamics are nevertheless able to provide solutions which are competitive with more sophisticated deterministic annealing algorithms, both in terms of quality of solutions and speed.

Before concluding, we would like to remark that our framework is more general than presented here, and we are presently employing it for solving more general subgraph isomorphism and relational structure matching problems. Preliminary experiments seem to indicate that local optima might represent a problem here, especially in matching very sparse or dense graphs. Escape procedures like those developed in (Bomze, 1997; Bomze et al., 1998) would therefore be helpful in these cases to avoid them. Nevertheless, local solutions in the continuous domain always have a meaningful interpretation in terms of maximal common subgraph isomorphisms, and this is one of the major advantages of the presented approach. We are currently conducting a thorough investigation, and we plan to present the results in a forthcoming paper. The approach is also being applied with success to the problem of matching hierarchical structures, with application to shape matching problems arising in computer vision (Pelillo, Siddiqi and Zucker, 1998).

Acknowledgments

This work was done while the author was visiting the Department of Computer Science at the Yale University, and was supported by Consiglio Nazionale delle Ricerche, Italy. The author would like to thank I. M. Bomze, A. Rangarajan, K. Siddiqi, and S. W. Zucker for many stimulating discussions and for providing comments on an earlier version of the manuscript, and the anonymous reviewers for constructive criticism.

References

Arora, S., Lund, C., Motwani, R., Sudan, M., and Szegedy, M. (1992). Proof verification and the hardness of approximation problems. In *Proc. 33rd Ann. Symp. Found. Comput. Sci.*, pages 14–23, Pittsburgh, PA.

Babai, L., Erdös, P., and Selkow, S. M. (1980). Random graph isomorphism. *SIAM J. Comput.*, 9(3):628–635.

Barrow, H. G. and Burstall, R. M. (1976). Subgraph isomorphism, matching relational structures and maximal cliques. *Inform. Process. Lett.*, 4(4):83–84.

Baum, L. E. and Eagon, J. A. (1967). An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology. *Bull. Amer. Math. Soc.*, 73:360–363.

Baum, L. E. and Sell, G. R. (1968). Growth transformations for functions on manifolds. *Pacific J. Math.*, 27(2):211–227.

Bellare, M., Goldwasser, S., and Sudan, M. (1995). Free bits, PCPs and non-approximability—Towards tight results. In *Proc. 36th Ann. Symp. Found. Comput. Sci.*, pages 422–431, Milwaukee, WI.

Bomze, I. M. (1997). Evolution towards the maximum clique. *J. Global Optim.*, 10:143–164.

Bomze, I. M., Budinich, M., Pelillo, M., and Rossi, C. (1998). Annealed replication: A new heuristic for the maximum clique problem. Technical Report CS98-10, Università di Venezia, Dipartimento di Matematica Applicata e Informatica (submitted for publication).

Bomze, I. M., Pelillo, M., and Giacomini, R. (1997). Evolutionary approach to the maximum clique problem: Empirical evidence on a larger scale. In Bomze, I. M., Csendes, T., Horst, R., and Pardalos, P. M., editors, *Developments in Global Optimization*, pages 95–108, Dordrecht, The Netherlands. Kluwer.

Boppana, R. B., Hastad, J., and Zachos, S. (1987). Does co-NP have short interactive proofs? *Inform. Process. Lett.*, 25:127–132.

Cabrales, A. and Sobel, J. (1992). On the limit points of discrete selection dynamics. *J. Econ. Theory*, 57:407–419.

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

Fisher, R. A. (1930). *The Genetical Theory of Natural Selection.* Oxford University Press, London, UK.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, San Francisco, CA.

Gaunersdorfer, A. and Hofbauer, J. (1995). Fictitious play, Shapley polygons, and the replicator equation. *Games Econ. Behav.*, 11:279–303.

Gibbons, L. E., Hearn, D. W., and Pardalos, P. M. (1996). A continuous based heuristic for the maximum clique problem. In Johnson, D. S. and Trick, M., editors, *Cliques, Coloring,* *and Satisfiability—Second DIMACS Implementation Challenge*, pages 103–124. American Mathematical Society.

Gibbons, L. E., Hearn, D. W., Pardalos, P. M., and Ramana, M. V. (1997). Continuous characterizations of the maximum clique problem. *Math. Oper. Res.*, 22(3):754–768.

Gold, S., and Rangarajan, A. (1996). A graduated assignment algorithm for graph matching. *IEEE Trans. Pattern Anal. Machine Intell.*, 18:377–388.

Grötschel, M., Lovász, L., and Schrijver, A. (1988). *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, Berlin.

Hastad, J. (1996). Clique is hard to approximate within $n^{1-\epsilon}$. In *Proc. 37th Ann. Symp. Found. Comput. Sci.*, pages 627–636, Burlington, VT.

Hofbauer, J. (1995). Imitation dynamics for games. Collegium Budapest, preprint.

Hofbauer, J. and Sigmund, K. (1988). *The Theory of Evolution and Dynamical Systems.* Cambridge University Press, Cambridge, UK.

Hofbauer, J. and Weibull, J. W. (1996). Evolutionary selection against dominated strategies. *J. Econ. Theory*, 71:558–573.

Hopfield, J. J. and Tank, D. W. (1985). "Neural" computation of decisions in optimization problems. *Biol. Cybern.*, 52:141–152.

Hummel, R. A. and Zucker, S. W. (1983). On the foundations of relaxation labeling processes. *IEEE Trans. Pattern Anal. Machine Intell.*, 5:267–287.

Johnson, D. S. (1988). The NP-completeness column: An ongoing guide. *J. Algorithms*, 9:426–444.

Kozen, D. (1978). A clique problem equivalent to graph isomorphism. *SIGACT News*, pages 50–52.

Kree, R. and Zippelius, A. (1988). Recognition of topological features of graphs and images in neural networks. *J. Phys. A: Math. Gen.*, 21:L813–L818.

Losert, V. and Akin, E. (1983). Dynamics of games and genes: Discrete versus continuous time. *J. Math. Biol.*, 17:241–251.

Luce, R. D. and Raiffa, H. (1957). *Games and Decisions.* John Wiley & Sons, New York.

Lyubich, Yu., Maistrowskii, G. D., and Ol'khovskii, Yu. G. (1980). Selection-induced convergence to equilibrium in a single-locus autosomal population. *Problems of Information Transmission*, 16:66–75.

Maynard Smith, J. (1982). *Evolution and the Theory of Games.* Cambridge University Press, Cambridge, UK.

Miller, D. A. and Zucker, S. W. (1991). Copositive-plus Lemke algorithm solves polymatrix games. *Oper. Res. Lett.*, 10:285–290.

Miller, D. A. and Zucker, S. W. (1992). Efficient simplex-like methods for equilibria of nonsymmetric analog networks. *Neural Computation*, 4:167–190.

Mjolsness, E., Gindi, G., and Anandan, P. (1989). Optimization in model matching and perceptual organization. *Neural Computation*, 1:218–229.

Motzkin, T. S. and Straus, E. G. (1965). Maxima for graphs and a new proof of a theorem of Turán. *Canad. J. Math.*, 17:533–540.

Palmer, E. M. (1985). *Graphical Evolution: An Introduction to the Theory of Random Graphs.* John Wiley & Sons, New York.

Pardalos, P. M. and Phillips, A. T. (1990). A global optimization approach for solving the maximum clique problem. *Int. J. Computer Math.*, 33:209–216.

Pardalos, P. M. and Xue, J. (1994). The maximum clique problem. *J. Global Optim.*, 4:301–328.

Pelillo, M. (1995). Relaxation labeling networks for the maximum clique problem. *J. Artif. Neural Networks*, 2:313–328.

Pelillo, M. (1997). The dynamics of nonlinear relaxation labeling processes. *J. Math. Imaging Vision*, 7:309–323.

Pelillo, M. and Jagota, A. (1995). Feasible and infeasible maxima in a quadratic program for maximum clique. *J. Artif. Neural Networks*, 2:411–420.

Pelillo, M., Siddiqi, K., and Zucker, S. W. (1998). Matching hierarchical structures using association graphs. In Burkhardt, H. and Neumann, B., editors, *Computer Vision—ECCV'98, Vol. II*, pages 3–16, Springer-Verlag, Berlin.

Rangarajan, A. (1997). Self-annealing: Unifying deterministic annealing and relaxation labeling. In Pelillo, M. and Hancock, E. R., editors, *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 229–244, Springer-Verlag, Berlin.

Rangarajan, A., Gold, S., and Mjolsness, E. (1996). A novel optimizing network architecture with applications. *Neural Computation*, 8:1041–1060.

Rangarajan, A. and Mjolsness, E. (1996). A lagrangian relaxation network for graph matching. *IEEE Trans. Neural Networks*, 7(6):1365–1381.

Rosenfeld, A., Hummel, R. A., and Zucker, S. W. (1976). Scene labeling by relaxation operations. *IEEE Trans. Syst. Man Cybern.*, 6:420–433.

Schöning, U. (1988). Graph isomorphism is in the low hierarchy. *J. Comput. Syst. Sci.*, 37:312–323.

Simić, P. D. (1991). Constrained nets for graph matching and other quadratic assignment problems. *Neural Computation*, 3:268–281.

Taylor, P. and Jonker, L. (1978). Evolutionarily stable strategies and game dynamics. *Math. Biosci.*, 40:145–156.

Waugh, F. R., and Westervelt, R. M. (1993). Analog neural networks with local competition. I. Dynamics and stability. *Phys. Rev. E*, 47(6):4524–4536.

Weibull, J. W. (1994). The 'as if' approach to game theory: Three positive results and four obstacles. *Europ. Econ. Rev.*, 38:868–881.

Weibull, J. W. (1995). *Evolutionary Game Theory.* MIT Press, Cambridge, MA.

Wilf, H. S. (1986). Spectral bounds for the clique and independence numbers of graphs. *J. Combin. Theory, Ser. B*, 40:113–117.

Yuille, A. L. and Kosowsky, J. J. (1994). Statistical physics algorithms that converge. *Neural Computation*, 6:341–356.