

N. S. Şengör · Y. Çakır · C. Güzeliş · F. Pekergin  
Ö. Morgül

## An analysis of maximum clique formulations and saturated linear dynamical network

Received: 18 March 1999/Accepted 26 April 1999

**Abstract** Several formulations and methods used in solving an NP-hard discrete optimization problem, maximum clique, are considered in a dynamical system perspective proposing continuous methods to the problem. A compact form for a saturated linear dynamical network, recently developed for obtaining approximations to maximum clique, is given so its relation to the classical gradient projection method of constrained optimization becomes more visible. Using this form, gradient-like dynamical systems as continuous methods for finding the maximum clique are discussed. To show the one to one correspondence between the stable equilibria of the saturated linear dynamical network and the minima of objective function related to the optimization problem, La Salle's invariance principle has been extended to the systems with a discontinuous right-hand side. In order to show the efficiency of the continuous methods simulation results are given comparing saturated the linear dynamical network, the continuous Hopfield network, the cellular neural networks and relaxation labelling networks. It is concluded that the quadratic programming formulation of the maximum clique problem provides a framework suitable to be incorporated with the continuous relaxation of binary optimization variables and hence allowing the use of gradient-like continuous systems which have been observed to be quite efficient for minimizing quadratic costs.

**Key words** Optimization · Maximum clique problem · Continuous methods · Gradient systems · Gradient-projection algorithm · Discontinuous differential equations · La Salle's invariance principle

### 1 Introduction

The maximum clique problem is to find a maximum complete subgraph of a graph. This graph theoretical problem is computationally equivalent to some other graph theoretical problems such as the maximum independent set and minimum vertex cover problem. The maximum clique problem and its equivalents are NP-hard optimization problems. However, it is essential to find the solution to them since these problems have theoretical and practical importance and are encountered in a diverse domain. The simplest method of finding the largest clique is to test all subsets of the vertices of a graph to see if they induce a complete graph. In the worst case, this method of solving the problem will give rise to a computing time that will exponentially grow with graph size. So in order to cope with its NP-hardness, different formulations and different algorithms have been used to solve the maximum clique problem and its equivalents. A complete review of the formulations and algorithms developed can be found in Pardalos and Xue (1994), Pelilo (1995), Jagota (1995) and Pekergin et al. (1998). In Pardalos and Rogers (1992), the problem is formulated as an unconstrained quadratic 0–1 program. In Pardalos and Rogers (1992), it is also given in a linear programming formulation with a unit simplex feasible region. In the papers that aim to solve maximum clique and equivalents in the neural network domain (Pelilo 1995, Jagota 1995, Funabihi et al. 1992, Grossman 1995, Şengör et al. 1998), energy descent optimizing dynamics are used. Yet another work is Pekergin et al. (1998) which benefits the saturated unstable linear dynamics. It is shown in Pekergin et al. (1998) that, for almost all initial conditions, any solution of this saturated

---

N. S. Şengör (✉) · Y. Çakır · C. Güzeliş  
Department of Electronics and Communication,  
Istanbul Technical University, Maslak 80626,  
Istanbul, Turkey  
e-mail: neslihan@ehb.itu.edu.tr,  
Tel.: + 90 212 285 36 19,  
Fax.: + 90 212 285 36 79

F. Pekergin  
LIPN-University Paris Nord, 93430-Villetaneuse, France

Ö. Morgül  
Bilkent University, Bilkent, Ankara, Turkey

This work was supported by TÜBİTAK-CNRS, Turkish and French Scientific and Technical Research Councils

linear gradient dynamical network defined on a closed hypercube reaches one of the vertices of the hypercube and any reached vertex corresponds to a maximal clique.

In recent years, there has been an interest in approaches based on continuous optimization. One of the main purposes of this paper is to show with a particular emphasis on the maximum clique problem, that gradient and gradient-like systems present efficient continuous solution methods for quadratic discrete optimization problems, and that the dynamical system theory provides a useful framework for analyzing such continuous methods and many others. Gradient dynamical systems can be described in a state equation from whose vector field is produced by the gradient of a scalar function, called energy. Energy descent and convergence properties of its completely stable equilibria to which trajectories converge correspond to local minima of the cost function. So called gradient-like systems covering quasi-gradient systems in Chiang and Chu (1996) and many dynamical neural networks (as special cases) which are in fact not gradient systems, but they also have the same kind of dynamics, can be used for minimizing cost functions with continuous optimization variables. It should be noted that the continuous Hopfield network (Hopfield 1982), the Grossberg neural network (Grossberg 1976), and the cellular neural network (Chua and Yang, 1988) are of gradient-like systems and are used for solving several optimization problems. Some variants of these networks such as the Continuous Hopfield Network (CHN) in Jagota (1995), the Grossberg type neural networks in Funabihi et al. (1992), the Relaxation Labeling Network (RLN) in Pelilo (1995), the Cellular Neural Network (CNN) in Şengör et al. (1998) and the Saturated Linear Dynamical Network (SLDN) (Pekergin et al. 1998) are used for finding approximate solutions to the maximum clique problem (Jagota 1995; Pekergin et al. 1998; Pardalos and Rogers 1992; Pardalos and Phillips 1990; Funabihi et al. 1992; Grossman 1995; Şengör et al. 1998). This paper analyzes the dynamics of gradient-like systems which, in the case of SLDN, gives rise to dynamical systems with a discontinuous right-hand side. The analysis shows that: 1. SLDN, which has been recently proposed (Pekergin et al. 1998) to obtain approximate solutions to the maximum clique problem and found to be successful, is, indeed, a continuous version of the classical gradient-projection algorithm of optimization theory. 2. La Salle's invariance principle can be extended to the systems with a discontinuous right-hand side, as a special case it is extended here for SLDN.

In Sect. 2, the maximum clique problem will be defined, different formulations and algorithms for solving the maximum clique problem will be described briefly. In Sect. 3, where the main contribution is given, the dynamics of gradient systems will be revisited. Then, the dynamics of saturated linear dynamical networks will be set up in a compact form and gradient-like systems will be discussed with the view of optimization. In this section, the stability analysis of gradient-like systems in the La Salle's

sense will be given by extending La Salle's (La Salle 1968) result on invariance principle to dynamical systems with discontinuous right-hand sides; hence it will be shown that there exists a one to one correspondence between the stable equilibria of SLDN and the minima of objective function. In Sect. 4, numerical results obtained using random graphs will be given for SLDN, CHN, CNN and RLN.

---

## 2 Comparison of maximum clique problem formulations

The maximum clique problem, which can be related to a number of different graph problems, is computationally intractable. Even to approximate it with certain bounds gives rise to the NP-hard problem. There is a large class of important problems that can be reduced to a maximum clique in principle. One example is the problem of finding the largest number of simultaneously satisfiable clauses (Crescenzi et al. 1991). Another class of problems that can be efficiently formulated as a maximum clique problem is the satisfiability of Boolean formulas (Garey and Johnson 1979). Applications of the maximum clique problem cover a large spectrum: pattern recognition, computer vision, information processing, cluster analysis information retrieval. First, definitions related to the maximum clique problem will be given. Also the adjacency matrix and characteristic vectors will be introduced and some results will be stated by a number of facts. Then, different formulations of the cost function for the problem will be given and the algorithms used for solving them will be compared.

In the following definitions, the graph is assumed to have no loop, no more than one edge associated to a vertex pair and has at least one edge.

**Definition Clique:** Let  $G = (V, E)$  be an undirected graph, where  $V$  is the set of vertices and  $E \subset V \times V$  is the set of edges. A subset  $S \subset V$  of vertices is called a clique if for every pair of vertices in  $S$  there is an edge in  $E$ , i.e., the subgraph introduced by  $S$  is complete.

**Definition Maximal Clique:** A maximal clique  $S$  is a clique of which proper extensions are not cliques, i.e. for any  $S'$  if  $S \subset S'$  and  $S \neq S'$  then  $S'$  is not a clique.

**Definition Maximum Clique:** A maximum clique of  $G$  is a clique for which the cardinality is maximum.

The maximum clique problem is to find the maximum cliques for a given graph.

For the formulations that will be introduced in the sequel the notion of an adjacency matrix and a characteristic vector is needed.

**Definition Adjacency Matrix:** Let  $G = (V, E)$  be an undirected graph. Let  $n = |V|$  be the number of vertices and let

$v_i \in V$ ,  $i = 1, 2, \dots, n$  denote the vertices.  $\mathbf{A} \in \{0, 1\}^{n \times n}$  is called the adjacency matrix of  $G$  iff  $\forall i, j \in \{1, 2, \dots, n\}$   $a_{ij} = a_{ji} = 1$  when  $(v_i, v_j) \in E$  and  $a_{ij} = a_{ji} = 0$  otherwise.

While  $\mathbf{A}$  denotes the adjacency matrix of  $G$ ,  $\bar{\mathbf{A}}$  denotes the adjacency matrix of the complement graph  $\bar{G}$ . Since  $G$  is an undirected graph and has no loops it follows that  $\mathbf{A}$  is a symmetric matrix with  $a_{ii} = 0$  for  $i \in \{1, 2, \dots, n\}$ .

**Definitions Characteristic Vector:** Let  $S \subset V$  be a subset of vertices,  $\mathbf{x}^s \in \{0, 1\}^n$  is the characteristic vector of  $S$  iff: 1)  $x_i^s = 1$  when  $v_i \in S$  2)  $x_i^s = 0$  when  $v_i \notin S$  for  $i \in \{1, 2, \dots, n\}$ .

Two results following these definitions will be given without proof by Facts 1 and 2:

**Fact 1**  $\bar{\mathbf{A}} \in \{0, 1\}^{n \times n}$  is an indefinite matrix.

**Fact 2**  $S$  is a maximal clique iff its characteristic vector  $\mathbf{x}^s$  satisfies the quadratic equation  $(\mathbf{x}^s)^T \bar{\mathbf{A}} (\mathbf{x}^s) = 0$ .

Fact 2 does not characterize the maximal clique  $S$  completely, but it shows that the adjacency matrix  $\bar{\mathbf{A}}$  is closely related to the characterization of clique.

The complete characterizations of the maximal cliques will be given by means of the following formulations. From the large number of max-clique problem formulations and algorithms only fundamental ones will be renewed. First the linear programming formulation, then the quadratic 0–1 programming formulation will be stated. Then different algorithms used and approaches dealing with the problem will be given for quadratic formulation.

## 2.1 Linear programming formulation

The maximum clique problem can be formulated as the simplest type of constrained optimization problems, i.e. linear programming, as follows:

minimize  $f_1(\mathbf{x}) = -\mathbf{e}^T \mathbf{x}$ , subject to  $x_i + x_j \leq 1$ ,

$$\forall (v_i, v_j) \in \bar{E} \quad \mathbf{x} \in \{0, 1\}^n$$

where,  $\mathbf{e} := [1, 1, \dots, 1]^T \in \mathbf{R}^n$ . A solution  $\mathbf{x}^*$  to this program defines a maximum clique  $S$  for  $G$  as follows: if  $x_i^* = 1$  then  $v_i \in S$  and if  $x_i^* = 0$  then  $v_i \notin S$  and the cardinality of  $S$ ,  $|S| = -f_1(\mathbf{x}^*)$ . This formulation can be carried to quadratic formulation which will be renewed in detail in the sequel by stating the constraints in the following way. Since for  $x_i, x_j \in \{0, 1\}$  and  $\forall (v_i, v_j) \in \bar{E}$ ,  $x_i + x_j \leq 1$ , holds iff  $x_i \cdot x_j = 0$ , the constraints in linear programming can be removed by adding quadratic terms to the objective function twice. It is well-known that the linear programming formulation of the maximum clique problem is not suitable for continuous methods since the continuous relaxation of the integer variables may lead to noninteger solutions.

## 2.2 Quadratic 0–1 programming

As mentioned in the previous part on linear programming formulation, the constrained linear optimization problem can be restated as unconstrained quadratic programming. In Pardalos and Rogers (1992) unconstrained quadratic 0–1 programming formulation is given not only for the maximum clique problem but also for the maximum independent set and minimum cover problems. Here only the formulation for the maximum clique will be renewed.

**Proposition 1** The maximum clique problem for the graph  $G$  is equivalent to solving the following quadratic 0–1 program. minimize  $f_2(\mathbf{x}) = \mathbf{x}^T [\bar{\mathbf{A}} - I] \mathbf{x}$ , such that  $\mathbf{x} \in \{0, 1\}^n$ .

The following theorem gives the correspondence between discrete local minima and maximal subgraphs.

**Theorem 1** Any  $\mathbf{x} \in \{0, 1\}^n$  that corresponds to a maximal subgraph of  $G$  is a discrete local minimum of  $f_2(\mathbf{x})$  in formulation given in Proposition 1. Conversely, any discrete local minimum of the function  $f_2(\mathbf{x})$  corresponds to a maximal subgraph of  $G$ .  $\square$

In Pardalos and Rogers (1992), a branch and bound algorithm which is based on this model is used. Branch and bound algorithms are set to find a global optimum by searching the entire branch and bound tree. This search is done by decomposing the given problem into subproblems.

Another quadratic 0–1 programming formulation (Pekergin et al. 1998), on which the SLDN is based, for the maximum clique problem is given as follows:

$$\min f_3(\mathbf{x}) := \mathbf{x}^T \bar{\mathbf{A}} \mathbf{x} - \mathbf{e}^T \mathbf{x}, \quad \mathbf{x} \in \{0, 1\}^n. \quad (1)$$

**Fact 3** Any  $\mathbf{x}^* \in \{0, 1\}^n$  is a (discrete) global minimum of  $f_3(\mathbf{x})$  given by the Eq. 1 iff the set  $S$  such that  $\mathbf{x}^S = \mathbf{x}^*$  is a maximum clique for  $G$ .

## 2.3 Motzkin-Straus formulation

In Pelilo (1995), and Pardalos and Phillips (1990), the maximum clique problem is formulated as an indefinite quadratic optimization problem but this time it is continuous and linearly constrained. In both of the papers (Pelilo 1995; Pardalos and Phillips 1990) mentioned, the methods are based on the Motzkin-Straus theorem given in Motzkin and Straus (1965). The formulation used in those papers is restated here:

$$\max f_4(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} \quad \mathbf{x} \in \kappa := \{\mathbf{x} \in \mathbf{R}^n | \mathbf{e}^T \mathbf{x} = 1, x_i \geq 0\}.$$

It has to be noted  $f_4(\mathbf{x})$  is indefinite and the feasible region is the unit simplex. The following theorem which relates the maximum clique problem to the above stated

formulation is reproduced in Pelilo (1995) and Pardalos and Phillips (1990) from the Motzkin-Straus theorem.

**Theorem 2** *If  $\alpha = \max f_4(\mathbf{x})$  over  $\kappa$  then  $G$  has a maximum clique  $S$  of size  $k = \frac{1}{1-2\alpha}$ . This maximum can be attained by setting  $x_i = \frac{1}{k}$  if  $v_i \in S$  and  $x_i = 0$  if  $v_i \notin S$ .  $\square$*

This theorem gives an approach to find the size of the maximum clique not the clique itself. The theorem given below is from Pardalos and Phillips (1990) and it presents a relationship between the set of distinct global maxima of  $f_4(\mathbf{x})$  over  $\kappa$  and the set of distinct maximum cliques of the graph  $G$ .

**Theorem 3** *Every distinct maximum clique of a graph  $G$  corresponds to a distinct global (hence local) maximum of the function  $f_4(\mathbf{x})$  over  $\kappa$ . The converse is false.  $\square$*

In Pardalos and Phillips (1990) to determine the vertices in the maximum clique an algorithm is presented, but it is reported in Pelilo (1995) and Pardalos and Phillips (1990) that the computational cost is excessive. Yet another approach based on the same formulation using Theorem 3 and a local version of it is given in Pelilo (1995). In this case, the formulation stated above is executed by a relaxation labeling network (RLN). Like other (Jagota 1995; Grossman 1995; Şengör et al. 1998) clique finding neural network models, the number of computational units used are as much as the number of vertices in the graph. Since this approach is suitable for parallel hardware implementation, the computational cost problem in Pardalos and Phillips (1990) is reduced. Here, the algorithm is based on the dynamics of the RLN which performs a gradient ascent search. If the solution obtained by the RLN has the particular form of  $x_i = \frac{1}{k}$  for some  $i$  and  $x_i = 0$  for the others, then this solution corresponds to a maximal clique. In this sense, the approach does not give rise to invalid solutions, but spurious solutions which are in the above particular form may arise. A benefit of the approach in Pelilo (1995) over the one in Pardalos and Phillips (1990), is that there is no need to calculate some parameters heuristically during the execution.

## 2.4 Hopfield network

Among the neural network based approaches used for the maximum clique problem (Pelilo 1995, Jagota 1995; Funabihi et al. 1992; Grossman 1995; Şengör et al. 1998), the one using Hopfield network (Jagota 1995) will be renewed here. The continuous dynamics and the energy function of the continuous Hopfield network are given below.

$$\dot{\mathbf{x}} = -\mathbf{x} + \mathbf{g}_\lambda(\mathbf{y}),$$

$$y_i = I + \sum_j w_{ij}x_j, \quad \mathbf{x} \in [0, 1]^n$$

$$E = -\frac{1}{2}\mathbf{x}^T\mathbf{W}\mathbf{x} - \mathbf{I}^T\mathbf{x} + \mathbf{e}^T\bar{\mathbf{g}},$$

$$\bar{\mathbf{g}} := \left[ \int_0^{x_1} g_\lambda^{-1}(x)dx \int_0^{x_2} g_\lambda^{-1}(x)dx \cdots \int_0^{x_n} g_\lambda^{-1}(x)dx \right],$$

where,  $\dot{\mathbf{x}}$  stands for the time-derivative of the state-vector  $\mathbf{x}$ .  $\mathbf{I} = [1, 1, \dots, 1]^T$  is the bias vector.  $\mathbf{W}$  is the weight matrix defined as:  $w_{ii} = 0$ ,  $w_{i,j} \in \{\rho, 1\}$  for all  $i \neq j$  with  $\rho < 0$ .  $w_{i,j} = w_{j,i} = 1$  iff there is an edge between the nodes  $i$  and  $j$ . Note that the weight matrix is not the adjacency matrix but closely related to it.  $\mathbf{g}_\lambda(\cdot) = [g_\lambda(\cdot), g_\lambda(\cdot), \dots, g_\lambda(\cdot)]^T$  is a separable function, each element  $g_\lambda(\cdot)$  of which is the sigmoidal function defined as:  $g_\lambda(x) = \frac{1}{1 + \exp^{-\lambda \cdot x}}$  with the gain factor  $\lambda$ . In this mentioned work (Jagota 1995), rather than considering the quadratic objective function and equating the energy function to this objective, the well-known Greedy algorithm is mapped into the dynamics of the CHN to find the maximum clique. In the suggested implementation of the CHN, the forward Euler method is used for the discretization, the number of iterations is chosen as the same as the graph at vertex number  $n$ , and furthermore  $\rho = -4n$ ,  $I = \frac{|\rho|}{4}$ . It is stated in Jagota (1965) that the stable equilibrium points of the considered CHN are maximal cliques of a graph  $G$  defining the weight matrix.

## 3 Gradient-like systems

A dynamical system of the form

$$\dot{\mathbf{x}} = -\nabla c(\mathbf{x}) \quad (2)$$

is called a gradient system and  $\nabla c(\mathbf{x})$  is the gradient vector of a scalar  $n$ -dimensional function  $c(\cdot)$ . The following well-known property of gradient systems makes them versatile in optimization problems (Hirsch and Smale 1974).

**Theorem 4** *The scalar function  $c(\cdot)$  does not increase along the trajectories, i.e.  $\dot{c}(\mathbf{x}(t)) \leq 0$  along the solutions  $\mathbf{x}(t)$  of (2). Moreover,  $\dot{c}(\mathbf{x}) = 0$  iff  $\mathbf{x}$  is an equilibrium of (2).  $\square$*

As Theorem 4 motivates, if the objective function of the optimization problem considered can be formulated as  $c(\cdot)$  in Eq. 2 which is also called “energy” due to the physical interpretation of Eq. 2 in many problems of mechanics etc., then the equilibrium points of the gradient system will coincide with the local minima of the objective function. As follows from the above discussion, the applicability of gradient systems in optimization problems is due to the one to one correspondence of the stable equilibria of the gradient system and the minima of the objective function. This approach to the optimization can be extended to the non-gradient but completely stable dynamical systems since every trajectory of a completely

stable dynamical system ends in one of the equilibrium points as in all gradient systems. If it is possible to formulate the objective function such that its minima coincides with the stable equilibrium points of a completely stable dynamical system, the dynamical system will solve the optimization problem since the minimum points will be its steady-state solutions. This is done to some extent in Chiang and Chu (1996) by generalizing gradient systems and forming so called quasi-gradient systems, and furthermore, as done here, by considering all gradient-like systems in the same context. It is shown in Chiang and Chu (1996), continuous versions of the methods as steepest descent, Newton. Branin can be implemented as quasi-gradient systems of the following form by choosing a suitable positive definite  $\mathbf{R}(\mathbf{x})$  matrix:  $\dot{\mathbf{x}} = -\mathbf{R}(\mathbf{x})^{-1} \cdot \nabla c(\mathbf{x})$ . In the sequel, it will be shown that SLDN (Pekergin et al. 1998), which is successfully used for solving the discrete optimization problem of maximum clique, constitutes an interesting class of gradient-like systems which are not gradient and also not quasi-gradient. To do that, a compact form is first presented for the SLDN originally proposed in Pekergin et al. (1998) to minimize a quadratic cost so its minimums are sought after continuous relaxation of variables on unit hypercube. From this compact form, it will be evident that the SLDN has a state equation form with a discontinuous right-hand side, but still solutions do exist and are uniquely defined as shown in Pekergin et al. (1998) and Hou and Michel (1998). An alternative (in a sense more rigorous) way to the derivation of complete stability of the SLDN in Pekergin et al. (1998) will be given here using La Salle's invariance principle (La Salle 1968). Since La Salle's invariance principle is derived for dynamical systems with continuous right-hand sides, an extension to the systems with discontinuous right-hand sides will be given.

In view of the gradient-like systems as solution methods for optimization problems, as will be evident by the given compact form, the most important fact about the SLDN is that the SLDN is indeed a continuous version of the well-known gradient projection method of the constrained optimization. This means that the SLDN and its variants (Jagota 1994) can be used not only for the maximum clique problem but also for other constrained optimization problems such as indefinite quadratic integer optimization problems and indefinite quadratic optimization defined over a polytope constraint set, etc.

SLDN is based on the 0–1 quadratic formulation in Eq. 1. The cost function  $E(\mathbf{x}) = f_3(\mathbf{x}) = \mathbf{x}^T \bar{\mathbf{A}}\mathbf{x} - \mathbf{e}^T \mathbf{x}$  is taken as “energy” hence the gradient-descent dynamics of the SLDN is obtained as  $\dot{\mathbf{x}} = -\frac{1}{2} \nabla E(\mathbf{x}) = \frac{1}{2} \mathbf{e} - \bar{\mathbf{A}}\mathbf{x}$ . Also, to handle the 0–1 integer constraint within these continuous dynamics, the  $\mathbf{x} \in \{0, 1\}^n$  integer constraint is relaxed to yield  $\mathbf{x} \in [0, 1]^n$ . Then, the solutions of the SLDN are restricted in the closed unit hypercube (Pekergin et al. 1998). Following this discussion the dynamics of the SLDN are derived in Pekergin et al. (1998)

as follows:

$$\dot{x}_i = \begin{cases} 0 & \text{if } x_i = 1 \text{ and } \frac{1}{2} - (\bar{\mathbf{A}}\mathbf{x})_i \geq 0 \\ 0 & \text{if } x_i = 0 \text{ and } \frac{1}{2} - (\bar{\mathbf{A}}\mathbf{x})_i \leq 0 \\ \frac{1}{2} - (\bar{\mathbf{A}}\mathbf{x})_i & \text{if otherwise} \end{cases} \quad (3)$$

The above dynamics show that, as long as the solutions are inside the hypercube, the trajectories follow the pure gradient descent direction, and that, as the solutions hit a surface of the hypercube, now the trajectories slide on the surface following the projected gradient descent direction. This fact explains that the SLDN behaves like the classical gradient projection algorithm of optimization (Motzkin and Straus 1965). So, the compact form introduced here is derived incorporating the following projection matrix  $\mathbf{P}_{I_a}$  (Luenberger 1973).

$$\mathbf{P}_{I_a} = [\mathbf{I} - \mathbf{B}_{I_a}^T (\mathbf{B}_{I_a} \mathbf{B}_{I_a}^T)^{-1} \mathbf{B}_{I_a}],$$

where,  $I_a(\mathbf{x})$  is the index set of active constraints which is the union of  $I_0$  and  $I_1$ , i.e.  $I_a := I_0 \cup I_1$ . The disjoint sets  $I_0, I_1$  indexing the active linear inequality constraints are defined as follows:

$$I_0(\mathbf{x}) := \{i \in N \mid x_i = 0 \text{ and } \frac{1}{2} - (\bar{\mathbf{A}}\mathbf{x})_i < 0\}$$

$$I_1(\mathbf{x}) := \{i \in N \mid x_i = 1 \text{ and } \frac{1}{2} - (\bar{\mathbf{A}}\mathbf{x})_i > 0\}.$$

$\mathbf{B}_{I_a}$  is an  $|I_a| \times n$  dimensional matrix whose  $j(i)$ 'th row,  $(\mathbf{B}_{I_a})_{j(i)}$  is defined as:

$$(\mathbf{B}_{I_a})_{j(i)} = \begin{cases} \mathbf{b}^T(i) & \text{if } i \in I_1 \\ -\mathbf{b}^T(i) & \text{if } i \in I_0. \end{cases}$$

Here,  $j(i) \in \{1, 2, \dots, |I_a|\}$  is an index used for renumbering the active constraints indexed by  $i$ . The  $j$ 'th row of  $\mathbf{B}_{I_a}$  depends on index  $i$ , so the number of rows of  $\mathbf{B}_{I_a}$  is as much as the number of active constraints.  $\mathbf{b}(i) \in \mathbf{R}^n$  is defined as:

$$(\mathbf{b}(i))_k = \begin{cases} 1 & \text{if } k = i \\ 0 & \text{if } k \neq i. \end{cases}$$

Now, the formed projection matrix  $\mathbf{P}_{I_a}$  can be given as follows:

$$[\mathbf{P}_{I_a}(\mathbf{x})]_{ij} = \begin{cases} 0 & i \neq j \\ 1 & i = j \in \bar{I}_a \\ 0 & i = j \in I_a. \end{cases}$$

It should be noted that, for any active set as  $I_a$ , this  $\mathbf{P}_{I_a}$  is symmetric and idempotent, i.e.  $\mathbf{P}_{I_a}^T = \mathbf{P}_{I_a} = \mathbf{P}_{I_a} \mathbf{P}_{I_a}$ . Now, the dynamics related to SLDN given in Eq. 3 can be written as follows:

$$\dot{\mathbf{x}} = -\mathbf{P}_{I_a}(\mathbf{x}) \nabla E(\mathbf{x}). \quad (4)$$

Even though the right-hand side of Eq. 4 is discontinuous in  $\mathbf{x}$ , it is known (Pekergin et al. 1998; Hou and Michel 1998) that, for any initial condition  $\mathbf{x}(0) \in [0, 1]^n$ , there exists a unique solution which is continuous, nondifferentiable but right differentiable with respect to time, and also is kept in the hypercube  $[0, 1]^n$ . The analysis, which will be given in the sequel, relies on the right-differentiability of the solutions. In Hou and Michel (1998), a model having the same dynamics as Eq. (4) is considered, and it is described with right-differentiable solutions. The concern of Hou and Michel (1998) is on the derivation of global asymptotical stability results which are useless for system 4 possessing multiple equilibria completely stable dynamics. The characteristics of the solutions of Eq. 4 are the same as those of the model in Hou and Michel (1998), so they will not be reconsidered here to avoid repetitions. However, the right-differentiability and uniqueness of the solutions of Eq. 4 lie on the following facts. 1) In every  $k$ -face of the hypercube  $[0, 1]^n$  with  $k \in \{0, 1, 2, \dots, n\}$ , the system 4 is equivalent to a linear state equation system. So, the solutions, starting at a  $k$ -face and staying there over a time interval, uniquely exist and are continuously differentiable in any degree. 2) The concatenation of these solution components yields a continuous solution since the projected gradients make right angles with the original gradients which are oriented towards the outside of the hypercube  $[0, 1]^n$  necessitating the projection. 3) The solutions may not be differentiable at the concatenation points. However, they are right-differentiable at these points since any concatenation point is, indeed, the starting point for another solution component which is differentiable. (See Hou and Michel (1998) for more details.)

The dynamical system 4 is not in the form of gradient system. Also, it is not a quasi-gradient system since  $\mathbf{P}_{f_a}(\mathbf{x})$  is not a positive definite matrix. But still, a discussion similar to the one given above dealing with the applicability of gradient systems in optimization problems can hold if it can be shown that the “energy” function  $\mathbf{E}(\mathbf{x})$  is nonincreasing along the trajectories of Eq. 4 and equilibria coincide with the local minima.

For this purpose, La Salle’s invariance principle which is originally given for systems having a continuous right-hand side will be extended here to the systems with a discontinuous right-hand side. Consider first Theorem 5 stating La Salle’s invariance principle for autonomous systems as  $\dot{\mathbf{x}} = \mathbf{h}(\mathbf{x})$  with a continuously differentiable right-hand side (Vidyasagar 1978).

**Theorem 5** *If there exists a continuously differentiable Lyapunov function  $V(\cdot): \mathbf{R}^n \rightarrow \mathbf{R}^1$  such that 1) the set  $\Omega_r = \{\mathbf{x} \in \mathbf{R}^n | V(\mathbf{x}) \leq r\}$  is bounded for some  $r > 0$ , 2)  $V(\cdot)$  is bounded below over such a set  $\Omega_r$ , and 3)  $\dot{V} \leq 0 \forall \mathbf{x} \in \Omega_r$ , then any solution  $\mathbf{x}(t, \mathbf{x}_0, 0)$ , starting from  $\mathbf{x}_0 = \mathbf{x}(0) \in \Omega_r$ , tends to the largest invariant set contained in  $S := \{\mathbf{x} \in \Omega_r | \dot{V}(\mathbf{x}) = 0\} \subset \Omega_r$ .*  $\square$

The largest invariant set mentioned in Theorem 5 consists of equilibrium points if the conditions of Theorem 6 (Chua and Wang 1978) are satisfied.

**Theorem 6** *The autonomous system  $\dot{\mathbf{x}} = \mathbf{h}(\mathbf{x})$  is completely stable, namely the invariant set which the trajectories tend to is made up of the equilibrium points if 1) the solutions of the system are bounded 2) there exists a continuously differentiable Lyapunov function  $V(\cdot)$  such that  $\dot{V} \leq 0 \forall \mathbf{x} \in \mathbf{R}^n$  except for the equilibrium points where it vanishes.*  $\square$

The system given in Eq. 4 has only bounded solutions, thus the first condition of Theorem 6 is satisfied. However, Theorem 6 can not be used to show the complete stability of Eq. 4 due to the discontinuous right-hand side of Eq. 4.

The key point in the proofs of Theorems 5–6 is the exploitation of the condition  $\dot{V}(\mathbf{x}) = [\nabla V(\mathbf{x})]^T \mathbf{h}(\mathbf{x}) \leq 0$ . This condition together with the other technical assumptions implies that the function  $V(\mathbf{x})$  is decreasing along trajectories until it reaches an equilibrium point where it takes a constant value. For the system in Eq. 4,  $\mathbf{x}$  is not a differentiable function of time  $t$ , special care has to be paid in using the time derivative of  $V(\cdot)$  and the chain rule  $\dot{V}(\mathbf{x}) = [\nabla V(\mathbf{x})]^T \dot{\mathbf{x}}$ . In the sequel, the right-derivative and the corresponding chain rule will be used to handle this problem. To handle another pathological case, in which the considered state equation system has a continuous right-hand side defined in an open set and the solutions are nondifferentiable even nonunique, La Salle’s paper (La Salle 1968) used a lower right-derivative for the Lyapunov function whose calculation, in general, requires the solutions to be known. Although the results of La Salle (1968) are given for a quite general class of systems, they can not be applied directly to the Eq. 4 since Eq. 4 has a discontinuous right-hand side defined over the closed (and also bounded) set  $[0, 1]^n$ . In the sequel, an invariance result for the discontinuous system 4 is given following a similar way to La Salle (1968). But, for system 4, it is known (Pekergin et al. 1998; Hou and Michel 1998) that the solutions are unique and the Lyapunov function candidate  $E(\mathbf{x})$ , which will be used, is continuously differentiable function. So, to use the lower right-derivative is restrictive. Instead, since the right limit of the solutions of Eq. 4 exists for all  $t$ , the right-derivative given below will be used, and then the right-derivative of the Lyapunov function along the trajectories will be calculated in terms of its gradient with respect to  $\mathbf{x}$  and the right-derivative of the solutions.

**Definition Right-Derivative:** *The right-derivative of a function  $\mathbf{x}(\cdot): \mathbf{R} \rightarrow \mathbf{R}^n$  is defined as  $\frac{d\mathbf{x}(t)}{dt^+} := \lim_{\Delta \rightarrow 0^+} \frac{\mathbf{x}(t+\Delta) - \mathbf{x}(t)}{\Delta}$  where  $\Delta \rightarrow 0^+$  means that  $\Delta$  approaches zero through positive values only.*

So in Lemma 1, a chain rule will be derived in the sense of right-differentiability.

**Lemma 1** *Consider the functions  $\psi(\cdot): D_\psi \subset [0, \infty) \rightarrow D_g \subset \mathbf{R}^n$  and  $g(\cdot): D_g \rightarrow \mathbf{R}$ . Let  $t \in \text{Int}(D_\psi)$  with  $\text{Int}$  stands*

for the set of interior points. Assume that  $g(\cdot)$  is continuously differentiable at  $\psi(t)$ , and  $\psi(\cdot)$  is right-differentiable at  $t$ . Then,  $g \circ \psi$  is right-differentiable at  $t$  and  $\frac{d(g \circ \psi)(t)}{dt^+} = [\nabla_{\psi} g(\psi)]^T \frac{d\psi(t)}{dt^+}$ .

**Proof**  $t \in \text{Int}(D_{\psi})$  implies  $\psi(t) \in \text{Int}(D_g)$ .  $\psi(\cdot)$  is right continuous due to the right-differentiability. Hence, by the definition of the right-derivative,

$$\frac{d(g \circ \psi)(t)}{dt^+} := \lim_{\Delta \rightarrow 0^+} \frac{g(\psi(t + \Delta)) - g(\psi(t))}{\Delta}.$$

As  $g(\cdot)$  is differentiable at  $\psi(t)$ , by mean value theorem,

$$g(\psi(t + \Delta)) - g(\psi(t)) = [\nabla g(\psi(\xi))]^T [\psi(t + \Delta) - \psi(t)]$$

for some  $\xi \in [t, t + \Delta]$ . Then,

$$\begin{aligned} \lim_{\Delta \rightarrow 0^+} \frac{g(\psi(t + \Delta)) - g(\psi(t))}{\Delta} \\ = \lim_{\Delta \rightarrow 0^+} [\nabla g(\psi(\xi))]^T \frac{[\psi(t + \Delta) - \psi(t)]}{\Delta}. \end{aligned}$$

Since  $g(\psi(\cdot))$  is differentiable, it can be written:

$$\lim_{\xi \rightarrow t^+} \nabla g(\psi(\xi)) = \nabla g(\psi(t)).$$

By the assumption of right-differentiability of  $\psi(\cdot)$ ,

$$\lim_{\Delta \rightarrow 0^+} \frac{\Psi(t + \Delta) - \psi(t)}{\Delta} = \frac{d\psi(t)}{dt^+}.$$

The limits of two sequences exist and the limit of the product sequence also exists, then this limit is equal to the product of the individual limits. This fact implies that  $g \circ \psi$  is right-differentiable:

$$\frac{d(g \circ \psi)(t)}{dt^+} = [\nabla_{\psi} g(\psi)]^T \frac{d\psi(t)}{dt^+}. \quad \square$$

So, Lemma 1 provides the needed chain rule as used in Lemma 2.

**Lemma 2** Consider system 4 and the corresponding “energy” function  $E(\mathbf{x}) = \mathbf{x}^T \bar{\mathbf{A}} \mathbf{x} - \mathbf{e}^T \mathbf{x}$ . Then,  $\frac{d(E \circ \mathbf{x})(t)}{dt^+} \leq 0 \forall \mathbf{x} \in [0, 1]^n$  and moreover it is equal to zero iff  $\mathbf{x}$  is an equilibrium point.

**Proof** The quadratic energy function  $E(\mathbf{x})$  is continuously differentiable with respect to  $\mathbf{x}$  and the solution  $\mathbf{x}(t)$  of Eq. 4 are unique and right-differentiable. So, due to Lemma 1,  $\frac{d(E \circ \mathbf{x})(t)}{dt^+} = [\nabla E(\mathbf{x})]^T \frac{d(\mathbf{x}(t))}{dt^+} = -[\nabla E(\mathbf{x})]^T P_{I_a}(\mathbf{x}) \nabla E(\mathbf{x})$ . Since  $P_{I_a}(\mathbf{x})$  is idempotent and symmetric,  $\frac{d(E \circ \mathbf{x})(t)}{dt^+} = -\|\nabla[E(\mathbf{x})]^T P_{I_a}(\mathbf{x})\|^2$  in terms of the Euclidean norm. Now,  $\frac{d(E \circ \mathbf{x})(t)}{dt^+}$  is equal to zero iff the vector  $P_{I_a}(\mathbf{x}) \nabla E(\mathbf{x})$  is

equal to zero. This specifies the equilibrium points of system 4.  $\square$

Lemmas 1 and 2 provide an extension of Theorem 6 to the considered discontinuous right-hand sided differential Eq. 4.

**Theorem 7 (Invariance Principle)** Consider the autonomous system 4 where the scalar function  $E(\mathbf{x}) = \mathbf{x}^T \bar{\mathbf{A}} \mathbf{x} - \mathbf{e}^T \mathbf{x}$ . Then, every trajectory that starts in  $[0, 1]^n$  ends in one of the equilibria, i.e. the system is completely stable.  $\square$

**Proof** For  $\mathbf{x}_0 := \mathbf{x}(0) \in [0, 1]^n$ , let  $\mathbf{x}(t, \mathbf{x}_0, 0)$  be the solution starting from  $\mathbf{x}_0$ . Due to the definition of the gradient projection operator, any such solution of Eq. 4 is bounded and is kept in the closed hypercube  $[0, 1]^n$ . Since the function  $E(\mathbf{x})$  is a continuous function, then it is bounded below over  $[0, 1]^n$ . It is known from Lemma 2 that  $\frac{dE(\mathbf{x}(t))}{dt^+} \leq 0 \forall \mathbf{x} \in [0, 1]^n$ . By the definition of the right-derivative,  $\frac{dE(\mathbf{x}(t))}{dt^+} \leq 0 \forall \mathbf{x} \in [0, 1]^n$  implies  $E(\mathbf{x}(t, \mathbf{x}_0, 0))$  is nonincreasing for all  $\mathbf{x} \in [0, 1]^n$ . This property together with the fact that  $E(\mathbf{x})$  is bounded below over  $[0, 1]^n$  yields:  $E(\mathbf{x}(t, \mathbf{x}_0, 0))$  converges to a limit  $E^*$ , i.e.

$$\lim_{t \rightarrow \infty} E(\mathbf{x}(t, \mathbf{x}_0, 0)) = E^*.$$

Due to the continuity of  $E(\mathbf{x})$ , and  $E(\mathbf{x}(t, \mathbf{x}_0, 0))$  goes to  $E^*$ ,  $\mathbf{x}(t, \mathbf{x}_0, 0)$  goes to the set  $\{\mathbf{x}^* | E(\mathbf{x}^*) = E^*\}$ . Such  $\mathbf{x}^*$ 's are, in fact, in the positive limit set  $L_+$  of the trajectory  $\mathbf{x}(t, \mathbf{x}_0, 0)$ . Since all sequences  $\{E(\mathbf{x}(t_n, \mathbf{x}_0, 0))\}_{n=1}^{\infty}$  have the same limit  $E^*$ , then  $E(\bar{\mathbf{x}}) = E^*$  for all  $\bar{\mathbf{x}} \in L_+$ . As known (Vidyasagar 1978), the positive limit set  $L_+$  is an invariant set, i.e.  $\mathbf{x}(t, \bar{\mathbf{x}}, 0) \in L_+$  for all  $\bar{\mathbf{x}} \in L_+$ . Hence,  $E(\mathbf{x})$  becomes constant along any trajectory starting at a point in  $L_+$ :

$$\frac{dE(\mathbf{x}(t, \bar{\mathbf{x}}, 0))}{dt^+} = 0 \quad \forall \bar{\mathbf{x}} \in L_+.$$

Now, by Lemma 2, the positive limit set,  $L_+$  of the trajectory  $\mathbf{x}(t, \mathbf{x}_0, 0)$  must consist of equilibrium points. By the definition of the equilibrium point and the uniqueness of the solutions,  $L_+$  for any trajectory contains the unique equilibrium point only.  $\square$

Theorem 7 shows that the system 4 has completely stable dynamics meaning that it does not possess oscillatory, chaotic or other exotic behaviours, so any trajectory of it converges to one of the equilibrium points. Due to the indefiniteness of the quadratic energy function  $E(\mathbf{x})$ , three different kinds of equilibrium points may coexist; namely stable, asymptotically stable, source or saddle type nonstable. Considering the relation between the gradient projection dynamics of Eq. 4 and the analyzed optimization problem, these equilibrium points correspond to, respectively, nonisolated local minima, isolated local minima, maxima or saddle points of  $E(\mathbf{x})$  over the constraint set  $[0, 1]^n$  (Pekergin et al. 1998). So, any locally

**Table 1** Average Cliques Sizes found for 100- and 500-graphs with density of 0.25, 0.50 and 0.75

V	Density	Overall Average			
		SLDN	CHN	CNN	RLN
100	0.25	4.83	4.48	4.176	5.16
	0.50	8.07	7.38	7.096	8.48
	0.75	15.05	13.87	13.562	16.31
500	0.25	5.8	5.617	5.155	6.02
	0.50	10.387	9.605	9.186	10.28
	0.75	21.022	19.443	19.318	-

asymptotically stable equilibrium point to which all trajectories starting at points in its vicinity constrained in  $[0, 1]^n$  converge, corresponds to a continuous isolated local minimum point of  $E(\mathbf{x})$ . It is proved in Pekergin et al. (1998) for  $E(\mathbf{x})$  with the considered  $\mathbf{A}$  that: 1) Any stable equilibrium point is also asymptotically stable, equivalently any local minimum point must be isolated. 2) The asymptotical equilibria necessarily occur on vertices of the hypercube  $[0, 1]^n$ . 3) The continuous local minima coincide with the discrete local minima of  $E(\mathbf{x})$  under the hypercube constraint  $[0, 1]^n$ . 4) Any converged vertex is actually a maximal clique. 5) For almost all initial states in the hypercube, any trajectory goes to one of the vertices corresponding to maximal clique. In other words, only the trajectories starting exactly on nonstable equilibria, which is known to be a zero measure, do not give a maximal clique. Therefore, calculating the steady-state solutions of the differential equation system 4 with  $E(\mathbf{x}) = \mathbf{x}^T \bar{\mathbf{A}} \mathbf{x} - \mathbf{e}^T \mathbf{x}$  is equivalent to finding maximal cliques of a graph given with the adjacency matrix  $\mathbf{A}$ .

#### 4 Simulation results

Performance of different clique finding methods are compared in Table 1 for random graphs of different vertex size and densities for SLDN, CHN, CNN, RLN. Average maximum cliques where the averages are taken over the test graphs generated with the same characteristics, i.e. the

vertex size and densities, are considered as a primary performance measure. Another performance measure is also considered, in which the averages are computed for the same test set but taking into account only the best results obtained on each graph is 5 and 10 independent runs with random initial conditions. This measure shows the ability of the methods to find different search directions when it is started from a different initial point. The results are summarized in Table 2. Since the RLN always starts with the same initial states, results for this method are not included.

#### 5 Conclusion

The maximum clique problem which is an NP-hard discrete optimization problem is reviewed here. Some basic formulations and methods used in solving this problem have been summarized, especially with emphasize on continuous methods. The main contribution of this work is given in Sect. 3. In this section, gradient and quasi-gradient systems were discussed, and it is shown that, even a system which has discontinuous right-hand side and hence can not be classified as both of these, a discussion similar to the above mentioned systems' optimizing dynamics can still be made. This discussion is carried on for a recently proposed dynamic optimizer, namely the saturated linear dynamical network, and to show gradient like (more precisely, completely stable) dynamics of such systems. La Salle's invariance principle is extended to the systems with a discontinuous right-hand side. Simulation results for continuous methods, namely the SLDN, CHN, CNN and RLN are given. It is concluded that gradient-like dynamical systems as continuous solution methods can be applied to the quadratic formulation of the maximum clique problem, offering good approximations.

#### References

Chiang HD, Chu CC (1996) A systematic search method for obtaining multiple local optimal solutions of nonlinear programming problems. IEEE Trans CASI 43, 2: 99-109

**Table 2** Averages over bests among 5 runs and bests among 10 runs for 100- and 500-graphs with density of 0.25, 0.50 and 0.75.

V	Density	Av. over Bests among 5 Runs			Av. over Bests among 10 Runs		
		SLDN	CHN	CNN	SLDN	CHN	CNN
100	0.25	5.21	4.58	4.48	5.30	4.62	5.14
	0.50	8.47	7.59	8.07	8.60	7.66	8.32
	0.75	15.63	14.24	14.69	15.76	14.40	15.18
500	0.25	6.476	6.169	6.285	6.80	6.38	6.551
	0.50	11.34	10.26	10.74	11.83	10.41	11.18
	0.75	22.437	20.593	21.67	23.00	20.875	22.20



- Chua LO, Yang Y (1988) Cellular neural networks: theory. *IEEE Trans Circuits Syst* 35: 1257–1272
- Chua LO, Wang NN (1978) Complete stability of autonomous reciprocal nonlinear networks. *Int J Circuit Theory and Applications*. 6: 211–241
- Crescenzi P, Fiorini C, Silvestri R (1991) A note on the approximation of the max-clique problem. *Inf Proc Lett*. 40 1: 1–5
- Funabihi N, Takefuji Y, Lee KC (1992) A neural network model for finding a near maximum clique. *Parallel and Distributed Comp* 14: 340–344
- Garey MR, Johnson DS (1979) *Computers and intractability: a guide to the theory of NP-completeness*. Freeman, New York
- Grosberg S (1976) Adaptive pattern classification and universal recording. *Biolog Cybernetics* 23: 121–134 and 187–202
- Grossmann Y (1995) Applying the INN model to the max clique problem. Second DIMACS Implementation Challenge, DIMACS series in Discrete Math and Theoretical Comp Science, American Math Society, Rhode Island
- Hirsch MW, Smale S (1974) *Differential equations. Dynamical systems and linear algebra*. Academic Press, New York
- Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc Math Acad Sci* 79: 2554–2558
- Hou L, Michel AN (1998) Asimptotic stability of systems with saturation constraints. *IEEE Trans Automatic Control* 43, 8: 1148–1154
- Jagota A (1994) A Hopfield style network with graph theoretic characterization. *J Artf Neur Net* 1: 145–167
- Jagota A (1995) Approximating maximum clique with a Hopfield network. *IEEE Trans Neur. Net* 6, 3: 724–735
- La Salle (1968) Stability theory for ordinary differential equations. *J. Differential Equations* 4: 57–65
- Luenberger DG (1973) *Introduction to linear and nonlinear programming*. Addison-Wesley, California
- Motzkin TS, Straus EG (1965) Maxima for graphs and new proof of a theorem of turan. *Canada J Math* 17: 533–540
- Pardolas PM, Phillips AT (1990) A global optimization approach for solving the maximum clique problem. *Int J Comp Math* 133: 209–216
- Pardalos PM, Rodgers GP (1992) A branch and bound algorithm for the maximum clique problem. *Comp Ops Res* 19 5: 363–375
- Pardolas P, Xue J (1994) The maximum clique problem. *J Global Optim* 4: 301–328
- Pekergin F, Morgül Ö and Güzeliş C (1998) A saturated linear dynamical net work for approximating maximum clique. *IEEE Trans on CAS I* (in press)
- Pelilo M (1995) Relaxation labeling networks for the maximum clique problem. *J Artf Neur Net* 2, 4: 313–328
- Şengör NS, Yalçın ME, Çakır Y, Üçer M, Güzeliş C, Pekergin F, Morgül Ö (1998) Solving maximum clique problem by cellular neural network. *Electronics Letters* 34, 15: 1504–1506
- Vidyasagar M (1978) *Nonlinear systems analysis*. Prentice Hall, New Jersey