An Analysis of Maximum Clique Formulations and Saturated Linear Dynamical Network

N. Şengör, Y. Çakır, C. Güzeliş, F. Pekergin *, Ö. Morgül **

İstanbul Technical University, Maslak 80626, İstanbul, Turkey
* LIPN - Université Paris-Nord, 93430-Villetaneuse, France
** Bilkent University, Bilkent, Ankara, Turkey

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

1 Introduction

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 maximum independent set problem and minimum vertex cover problem. Maximum clique problem and its equivalents are NP-hard optimization problems however, it is indispensable to find solution to them since these problems have theoretical and practical importance and are encountered in a diverse domain. The simplest idea to find the largest clique is to test all subsets of the vertices of graph to see if they induce a complete graph. In the worst case, this idea in solving the problem will give rise to a computing time exponentially growing 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 [1,2,3,4]. In [5], the problem is formulated as an unconstrained quadratic 0-1 program. Again in [5], it is also given in linear programming formulation with unit simplex feasible region. In the papers that

aim to solve maximum clique and equivalents in the neural network domain [2,3,7,8,9], energy descent optimizing dynamics is used. Yet another work is [4] which benefits the saturated unstable linear dynamics. It is shown in [4] that, for almost all initial conditions, any solution of this saturated 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 the last years, there is an interest on approaches based on continuous optimization. One of the main purposes of this paper is to show with a particular emphasize on maximum clique problem that gradient and gradient-like systems present efficient continuous solution methods for quadratic discrete optimization problems, and dynamical system theory provides a usefull framework for analyzing such continuous methods and many others. Gradient dynamical systems can be described in a state equation form whose vector field is produced by the gradient of a scalar function, called energy. Energy descent completely stable dynamics makes them suitable for minimizing a cost function such that the stable equilibria to which trajectories converge correspond to local minima of the cost function. So called gradient-like systems covering quasi-gradient systems in [10] and many dynamical neural networks as special cases which are, in fact, not gradient systems but they also have the same kind of dynamics and hence can be used for minimizing cost functions with continuous optimization variables. It should be noted that continuous Hopfield network [11], Grossberg neural network [12], cellular neural network [13] are of gradient-like systems and are used for solving several optimization problems. Some variants of these networks such as Continuous Hopfield Network (CHN) in [3], Grossberg type neural networks in [7], Relaxation Labeling Network (RLN) in [2], Cellular Neural Network (CNN) in [9] and Saturated Linear Dynamical Network (SLDN) [4] are used for finding approximate solutions to the maximum clique problem [3-9]. This paper analyzes the dynamics of gradient-like systems which, in the case of SLDN, gives rise to dynamical systems with discontinuous right-hand side. The analysis shows that: i) SLDN, which is recently proposed [4] to obtain approximate solutions to the maximum clique problem and found to be succesful, is, indeed, a continuous version of the classical gradient-projection algorithm of optimization theory. ii) La Salle's invariance principle can be extended to the systems with discontinuous right-hand side, as a special case it is extended here for SLDN.

In Section 2, maximum clique problem will be defined, different formulations and algorithms in solving maximum clique problem will be described briefly. In Section 3, where the main contribution is given, dynamics of gradient systems will be revisited first. Then, the dynamics of saturated linear dynamical network will be set up in a compact form and gradient-like systems will be discussed in 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 [14] result on invariance principle to dynamical system with discontinuous right-hand sides; hence it will be shown that there exists a one to one corespondence between the stable equilibria of SLDN and the minima of objective function. In Section 4, numerical results obtained using random graphs will be given for SLDN, CHN, CNN and RLN.

2 Comparison of Maximum Clique Problem Formulations

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 NP-hard problem. There is a large class of important problems that can be reduced to maximum clique in principle. One example is the problem of finding the largest number of simultaneously satisfiable clauses [15]. Another class of problems that can be efficiently formulated as a maximum clique problem is the satisfiability of Boolean formulas [16]. Applications of maximum clique problem cover a large spectrum: pattern recognition, computer vision, information processing, cluster analysis information retrival. First, definitions related to maximum clique problem will be given. Also adjacency matrix and characteristic vector 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 have 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. □

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 adjacency matrix and characteristic vector is needed.

Definition *"Adjacency Matrix"* : Let $G = (V, E)$ be an undirected graph. Let $n = |V|$ the number of vertices and let $v_i \in V$, $i = 1, 2, .., n$ denote the vertices $\mathbf{A} \in \{0, 1\}^{n \times n}$ is the adjacency matrix of G, i.e., for $i, j = 1, 2, .., n, a_{ij} = a_{ji} = 1$ iff $(v_i, v_j) \in E$. □

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

Definition *"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: i) $x_i^s = 1$ iff $v_i \in S$ ii) $x_i^s = 0$ iff $v_i \notin S$ for $i = 1, 2, .., n$. □

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

Fact 1: $\bar{\mathbf{A}} \in \{0,1\}^{nxn}$ 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 maximal clique S completely, but it shows that 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 linear programming formulation, then quadratic 0-1 programming formulation will be stated. Then different algorithms used and approaches dealing with the problem will be given for quadratc 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:

$$\textit{minimize} \quad f_1(\mathbf{x}) = -\mathbf{e}^T\mathbf{x}, \quad \textit{subject to} \quad x_i + x_j \leq 1, \quad \forall (v_i, v_j) \in \overline{E} \quad \mathbf{x} \in \{0,1\}^n$$

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 $\forall (v_i, v_j) \in \overline{E}$, $x_i + x_j \leq 1$, 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 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, constrained linear optimization problem can be restated as unconstrained quadratic programming. In [5] unconstrained quadratic 0-1 programming formulation is given not only for maximum clique problem but also for maximum independent set and minimum cover problems. Here only the formulation for 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. $\textit{minimize}\ f_2(\mathbf{x}) = \mathbf{x}^T[\overline{\mathbf{A}} - I]\mathbf{x}$, 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. □

A branch and bound algorithm which is based on this model is used in [5]. Branch and bound algorithms are set to find a global optimum by searching entire branch and bound tree. This search is done by decomposing the given problem into subproblems.

Another quadratic 0-1 programming formulation [4], on which the SLDN is based, for the maximum clique problem is given as follows.

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

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

2.3 Motzkin-Straus Formulation

In [2,6], 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 [2] and [6] to be mentioned, the methods are based on Motzkin-Straus theorem [13]. The formulation used in those papers is restated here:

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

It has to be noted $f_4(\mathbf{x})$ is indefinite and feasible region is the unit simplex. The following theorem which relates the maximum clique problem to the above stated formulation is reproduced from Motzkin-Straus theorem [2,6].

Theorem 2: If $\alpha = max\; f_4(\mathbf{x})$ over κ 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 $v_i \notin S$. □

This theorem gives an approach to find the size of the maximum clique not the clique itself. Theorem given below is from [6] and it presents a relationship between the set of distinct global maxima of $f_4(\mathbf{x})$ over κ 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 κ . The converse is false. □

In [6] to determine the vertices in the maximum clique an algorithm is presented, but it is reported in [2,6] 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 [2]. In this case, the formulation stated above is executed by relaxation labeling network (RLN). Like other clique finding neural network models [3,8,9], number of the 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 [6] is reduced. Here, the algorithm is based on the dynamics of the RLN which performs a gradient ascent search. If the solution obtained by 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 [2] over the one in [6], there is no need to calculate some parameters heuristically during the execution.

2.4 Hopfield Network

Among the neural network based approach used in maximum clique problem [2,3,7-9], the one using Hopfield Network [3] will be renewed here. The continuous dynamics and the energy function of the continuous Hopfield network are given below.

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

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

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

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

Where, $\dot{\mathbf{x}}$ stands for the time-derivative of the state-vector $\mathbf{x}$. $\mathbf{I} = [1, 1, ..., 1]^T$ 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. $\mathbf{g}_\lambda(\cdot) = [g_\lambda(\cdot), g_\lambda(\cdot), ..., g_\lambda(\cdot)]^T$ 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 λ. In this mentioned work [3], 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 CHN to find maximum clique. In the suggested implementation of the CHN, forward Euler method is used for the discretization, the number of iterations is chosen as the same with the graph vertex number n, and furthermore $\rho = -4 \cdot n$, $I = \frac{|\rho|}{4}$. It is stated in [3] 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

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

is called 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 [19] make them

versatile in optimization problems.

Theorem 4: $\dot{c}(\mathbf{x}) \leq 0$ for all $\mathbf{x} \in R^n$, $\dot{c}(\mathbf{x}) = 0$ iff $\mathbf{x}$ is an equilibrium of (2). $\Box$

As the theorem (4) motivates, if the objective function of the optimization problem considered can be formulated as $c(\cdot)$ in (2) which is also called "energy" due to the physical interpretation of (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 [10] 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 [10], 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 [4], which is succesfully 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, first a compact form is presented for SLDN originally proposed in [4] to minimize a quadratic cost so its minimums are seeked after continuous relaxation of variables on unit hypercube. From this compact form, it will be evident that SLDN has a state equation form with discontinuous right-hand side, but still the solutions do exist and are uniquely defined as shown in [4,21]. An alternative (in a sense more rigorous) way to the derivation of complete stability of SLDN in [4] will be given here using La Salle's invariance principle [14]. 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 SLDN is that SLDN is indeed a continuous version of the well-known gradient projection method of the constrained optimization. This means that SLDN and its variants [18] 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 (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 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 0-1 integer constraint within this continuous dynamics, $\mathbf{x} \in \{0,1\}^n$ integer constraint is relaxed to yield $\mathbf{x} \in [0,1]^n$.

Then, the solutions of SLDN are restricted in the closed unit hypercube [4]. Following this discussion the dynamics of SLDN is derived in [4] as follows:

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

The above dynamics shows 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 SLDN behaves like the classical gradient projection algorithm of optimization [17]. So, the compact form introduced here is derived incorporating the following projection matrix $\mathbf{P}_{I_a}$ [20].

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

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:

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

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

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

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

Here, $j(i) \in \{1, 2, ..., |I_a|\}$ 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:

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

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

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

It should be noted that, for any active set as I_a, this $\mathbf{P}_{I_a}$ is idempotent, i.e, $\mathbf{P}_{I_a} = \mathbf{P}_{I_a}\mathbf{P}_{I_a}$, nonexpansive and furthermore $||\mathbf{P}_{I_a}|| \leq ||x||$. Now the dynamics related to SLDN given in (3) can be written as follows:

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

Eventhough the right-hand side of Equation (4) is discontinuous in $\mathbf{x}$, it is known [4,21] 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 [21], a model having the same dynamics with (4) is considered, and it is described to have right differentiable solutions. To avoid repetitions, the characteristics of the solutions of (4) will not be taken into the reconsideration. But, the concern of [21] is on the derivation of global asymptotical stability results which are useless for System (4) possessing multiple equilibria completely stable dynamics.

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

For this purpose, La Salle's invariance principle which is originally given for systems having continuous right-hand side will be extended here to the systems with 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 continuously differentiable right-hand side [23].

Theorem 5: If there exists a continuously differentiable Lyapunov function $V(\cdot) : \mathbf{R}^n \to \mathbf{R}^1$ such that i) the set $\Omega_r = \{\mathbf{x} \in \mathbf{R}^n | V(\mathbf{x}) \leq r\}$ is bounded for some $r > 0$, ii) $V(\cdot)$ is bounded below over such a set Ω_r, and iii) $\dot{V} \leq 0 \ \ \forall \mathbf{x} \in \Omega_r$, 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 $S := \{\mathbf{x} \in \Omega_r | \dot{V}(\mathbf{x}) = 0\} \subset \Omega_r$. □

The largest invariant set mentioned in Theorem 5 consists of equilibrium points if the conditions of Theorem 6 [22] 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 i) the solutions of the system are bounded, ii) 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. □

The system given in (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 (4) due to the discontinuous right-hand side of (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 reaches an equilibrium point where it takes a constant value. For the system in (4), $\mathbf{x}$ is not a differantiable 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, right derivative and the

corresponding chain rule will be used to handle this problem. To handle another pathological case, La Salle's paper [14] used lower right derivative since there, the solutions were not unique for the considered continuous right-hand sided differential equation. For the state equation (4), it is known [4,21] that the solutions are unique, so to use lower right derivative is too restrictive. Instead, since the right limit of the right-hand side of (4) exist for all t, the right derivative [1] will be used. So in Lemma 1, chain rule will be derived in the sense of right differentiability.

Lemma 1 : Consider the functions $\psi(\cdot) : D_\psi \subset [0,\infty) \to D_g \subset R^n$ and $g(\cdot) : D_g \subset R^n \to R$. Let $t \in Int(D_\psi)$. Assume that $g(\cdot)$ is continuously differantiable at $\psi(t)$, and $\psi(\cdot)$ is right differantiable 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 :

$t \in Int(D_\psi)$ implies $\psi(t) \in int(D_g)$. Because, $\psi(\cdot)$ is right continuous due to the right differentiability. Hence, by the definition of right derivative,

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

As $g(\cdot)$ is differantiable 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,

$$\lim_{\Delta\to 0^+} \frac{g(\psi(t+\Delta)) - g(\psi(t))}{\Delta} = \lim_{\Delta\to 0^+} [\nabla g(\Delta(\xi))]^T \frac{[\psi(t+\Delta) - \psi(t)]}{\Delta} \quad .$$

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

$$\lim_{\Delta\to 0^+} \nabla g(\psi(\xi)) = \nabla g(\psi(\xi)) \quad .$$

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

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

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 differantiable:

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

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

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

Lemma 2 : Consider the system (4) and the corresponding "energy" function $E(\mathbf{x}) = \mathbf{x}^T \bar{\mathbf{A}} \mathbf{x} - e^T \mathbf{x}$. Then, $\frac{d(E \circ \mathbf{x})(t)}{dt^+} \leq 0 \quad \forall \mathbf{x} \in [0,1]^n$ 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 solutions $\mathbf{x}(t)$ of (4) are unique and right differentiable. So, due to Lemma 1, $\frac{d(E \circ \mathbf{x}(t))}{dt^+} = [\nabla E(\mathbf{x})]^T \frac{d(\mathbf{x}(\mathbf{t}))}{dt^+} = -[\nabla E(\mathbf{x})]^T P_{I_a}(\mathbf{x}) \nabla E(\mathbf{x})$. Since $P_{I_a}(\mathbf{x})$ is idempotent, $\frac{d(E \circ \mathbf{x}(t))}{dt^+} = -||\nabla [E(\mathbf{x})]^T P_{I_a}(\mathbf{x})||^2$ 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). □

Lemmas 1 and 2 provides an extension of Theorem 6 to the considered discontinuous right-hand sided differential equation (4).

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

Proof : For $\mathbf{x}_0 := \mathbf{x}(\mathbf{0}) \in [0,1]^n$, let $\mathbf{x}(\mathbf{t}, \mathbf{x_0}, \mathbf{0})$ be the solution starting from $\mathbf{x}_0$. Due to the definition of the gradient projection operator, any such solution of (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.

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

Due to the continuity of $E(\mathbf{x})$, when $E(\mathbf{x}(t, \mathbf{x}_0, 0))$ goes to E^*, $\mathbf{x}(t, \mathbf{x}_0, 0)$ goes to some constant vector $\mathbf{x}^*$. $\mathbf{x}^*$ is, indeed, in the positive limit set L_+ of the trajectory $\mathbf{x}(t, \mathbf{x}_0, 0)$. Since all sequences $\{E(\mathbf{x}(t_n, \mathbf{x}_0, 0))\}_{n=1}^{\infty}$ has the same limit E^*, then $E(\bar{\mathbf{x}}) = E^*$ for all $\bar{\mathbf{x}} \in L_+$. As known [23], 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_+:

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

Now, by Lemma 2, the positive limit set L_+ of the trajectory $\mathbf{x}(t, \mathbf{x}_0, 0)$ must consist of equilibrium points which are necessarily unique. □

Theorem 7 shows that the system (4) has completely stable dynamics, meaning any trajectory of it converges to an equilibrium point. These equilibrium points are, indeed, local minima, maxima and saddle points of the quadratic energy function $E(\mathbf{x}$ over the constraint set $[0,1]^n$ [4]. So, any stable equilibrium point, which all trajectories starting at points in some neighborhood constrained to $[0,1]^n$ converge to, corresponds to a continuous local minimum point of $E(\mathbf{x}$. It is proved in [4] that these continuous

minima coincide with the discrete minima of $E(\mathbf{x}$ under the hypercube constraint $[0,1]^n$. Therefore, calculating the steady-state solutions of the differential equation system (4) with $E(\mathbf{x}) = \mathbf{x}^T\bar{\mathbf{A}}\mathbf{x} - e^T\mathbf{x}$ is equivalent to finding maximal cliques of a graph given with the adjacency matrix $\mathbf{A}$.

4 Experimental Results

Performance of different clique finding methods were compared in Table 1 for random graphs of different vertex size and densities for SLDN, CHN, CNN, RLN. Average maximum cliques where the averages is taken over the test graphs generated with the same characteristics, i.e., the vertex size and densities, is 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 in 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 different initial point. The results are summarized in Table 2. Since the RLN starts always with the same initial states results for this method are not included.

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
	0.5	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
	0.5	10.387	9.605	9.186	10.28
	0.75	21.022	19.443	19.318	-

Table 2. Averages over Bests among 5 runs and Bests among 10 runs for a 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	4.84	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

5 Conclusion

Maximum clique problem which is an NP-hard discrete optimization problem is reviewed here. Some basic formulations and methods used in solving this problem are summarized, especially with emphasize on continuous methods. The main contribution of this work is given in Section 3. In this section, gradient and quasi-gradient systems are 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, still a discussion similar to the above mentioned systems' optimizing dynamics can be given. 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 discontinuous right-hand side. Experimental results for continuous methods, namely 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 with offering good approximations.

References

1. P.Pardolas, J.Xue, The Maximum Clique Problem *" J.Global Optim"* . Vol.4, pp.301-328. (1994)
2. M.Pelilo, Relaxation Labeling Networks for the Maximum Clique Problem *"J. of Artf. Neur. Net."* Special Issue on Neural Networks for Optimization, (1995)
3. A.Jagota, Approximating Maximum Clique with a Hopfield Network *"IEEE Trans. Neural Net."* , Vol.6, No.3, pp. 724-735. (1995)
4. F.Pekergin, Ö.Morgül and C.Güzeliş A Saturated Linear Dynamical Network for Approximating Maximum Clique, *" IEEE Trans. on CAS Part-I"* , (Accepted.) (1998)
5. P.M.Pardalos and G.P.Rodgers, A Branch and Bound Algorithm for the Maximum Clique Problem, *"Computers Ops. Res."* , Vol.19, No.5, pp.363-375. (1992)
6. P.M.Pardolas, A.T.Phillips, A Global Optimization Approach for Solving the Maximum Clique Problem, *"Intern.J. Copm. Math."* , Vol. 133, pp.209-216. (1990)
7. N.Funabihi, Y.Takefuji, K.C.Lee, A Neural Network Model for Finding a Near Maximum Clique, *"J. Parallel and Distributed Comp."* , Vol.14, pp.340-344. (1992)
8. Y.Grossman, Applying the INN Model to the Max Clique Problem, Second DIMACS Implementation Challenge, DIMACS series in Discrete Math. and Theoretical Comp. Science. (1995)
9. N.S.Şengör, M.E.Yalçın, Y.Çakır, M.Üçer, C.Güzeliş, F.Pekergin, Ö.Morgül, Solving maximum clique problem by cellular neural network, *"Electronics Letters"*, Vol.34, No.15, pp.1504-1506. (1998)

10. H.D.Chiang, C.C.Chu, A Systematic Search Method for Obtaining Multiple Local Optimal Solutions of Nonlinear Programming Problems, *"IEEE Trans. CAS-I "*, Vol.43, No.2, pp.99-109. (1996)

11. J.J.Hopfield Neuarl Networks and Pysical Systems With Emergent Collective Computational Abilities, *"Proc. Math. Acad. Sci."*, Vol.79, pp.2554-2558. (1982)

12. S.Grosberg, Adaptive Pattern Classification and Universal Recording, *"Biolog. Cybernetics"*, Vol. 23, pp. 121-134 and pp.187-202. (1976)

13. L.O.Chua, Y.Yang, Cellular Neural Networks: Theory, *"IEEE Trans. Circuits Syst. "*, Vol. 35, pp.1257-1272. (1988)

14. La Salle, Stability Theory for Ordinary Differantial Equations, *"Journal of Differential Equations"*, Vol.4, pp.57-65. (1968)

15. P.Crescenzi, C.Fiorini, R.Silvestri, A Note on the Approximation of the Max-clique Problem, *"Inf. Proc.Lett."*, Vol.40, No.1, pp.1-5. (1991)

16. M.R.Garey, D.S.Johnson, Computers and Intractibility: A Guide to the Theory of NP-completeness, Newyork,Freeman. (1979)

17. T.S.Motzkin,E.G.Straus, Maxima for Graphs and a New Proof of a Theorem of Turan, *"Canada J. Math"*, vol.17, pp.533-540. (1965)

18. A.Jagota, A Hopfield Style Network with Graph Theoretic Characterization, *"J. Artf. Neur. Net."*, Vol.1, pp.145-167. (1994)

19. M.W.Hirsch, S.Smale, Differential Equations, Dynamical Systems and Linear Algebra, Akademic Press. (1974)

20. D.G.Luenberger, Introduction to Linear and Nonlinear Programming, Addison-Wesley Publishing Com. (1973)

21. L.Hou, A.N.Michel, Asimptotic Stability of Systems with Saturation Constraints, *"IEEE Trans. Automatic Control"*, Vol.43, No.8. pp.1148-1154. (1998)

22. L.O.Chua, N.N.Wang, Complete Stability of Autonomus Reciprocal Nonlinear Networks, *"Int. J. Circuit Theory and Applications"*, Vol.6, pp.211-241. (1978)

23. M.Vidyasagar, Nonlinear Systems Analysis, *"Prentice Hall "*. (1978)