Continuous-time relaxation labeling processes

Andrea Torsello[a], Marcello Pelillo[b],*

[a]*Micro Strategy, 8000 Towers Crescent Drive, Vienna, VA 22182, USA*
[b]*Dipartimento di Informatica, Università Ca' Foscari di Venezia, Via Torino 155, 30172 Venezia Mestre, Italy*

Abstract

We study the dynamical properties of two new relaxation labeling schemes described in terms of differential equations, and hence evolving in continuous time. This contrasts with the customary approach to defining relaxation labeling algorithms which prefers discrete time. Continuous-time dynamical systems are particularly attractive because they can be implemented directly in hardware circuitry, and the study of their dynamical properties is simpler and more elegant. They are also more plausible as models of biological visual computation. We prove that the proposed models enjoy exactly the same dynamical properties as the classical relaxation labeling schemes, and show how they are intimately related to Hummel and Zucker's now classical theory of constraint satisfaction. In particular, we prove that, when a certain symmetry condition is met, the dynamical systems' behavior is governed by a Liapunov function which turns out to be (the negative of) a well-known consistency measure. Moreover, we prove that the fundamental dynamical properties of the systems are retained when the symmetry restriction is relaxed. We also analyze the properties of a simple discretization of the proposed dynamics, which is useful in digital computer implementations. Simulation results are presented which show the practical behavior of the models. © 2000 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Relaxation labeling processes; Differential equations; Dynamical systems; Consistency

1. Introduction

Relaxation labeling processes are a popular class of parallel, distributed computational models aimed at solving (continuous) constraint satisfaction problems, instances of which arise in a wide variety of computer vision and pattern recognition tasks [1,2]. Almost invariably, all the relaxation algorithms developed so far evolve in discrete time, i.e., they are modeled as difference rather than as differential equations. The main reason for this widespread practice is that discrete-time dynamical systems are simpler to program and simulate on digital

computers. However, continuous-time dynamical systems are more attractive for several reasons. First, they can more easily be implemented in parallel, analog circuitry (see, e.g., Ref. [3]). Second, the study of their dynamical properties is simplified thanks to the power of differential calculus, and proofs are more elegant and more easily understood. Finally, from a speculative standpoint, they are more plausible as models of biological computation [4].

Recently, there has been some interest in developing relaxation labeling schemes evolving in continuous time. In particular, we cite the work by Stoddart [5] motivated by the Baum–Eagon inequality [6], and the recent work by Li et al. [7] who developed a new relaxation scheme based on augmented Lagrangian multipliers and Hopfield networks. Yu and Tsai [8] also used a continuous-time Hopfield network for solving labeling problems. All these studies, however, are motivated by the assumption that the labeling problem is formulated as an

* Corresponding author. Tel.: + 39-41-2908440; fax: + 39-41-2908419.

E-mail addresses: atorsello@strategy.com (A. Torsello), pelillo@dsi.unive.it (M. Pelillo).

energy-minimization problem, and a connection to standard theories of consistency [9] exists only when the compatibility coefficients are assumed to be symmetric. This is well-known to be a restrictive and unrealistic assumption. When the symmetry condition is relaxed the labeling problem is equivalent to a variational inequality problem, which is indeed a generalization of standard optimization problems [9].

In this paper, we study the dynamical properties of two simple relaxation labeling schemes which evolve in continuous time, each being described in terms of a system of coupled differential equations. The systems have been introduced in the context of evolutionary game theory, to model the evolution of relative frequencies of species in a multi-population setting [10], and one of them has also recently been proposed by Stoddart et al. [5], who studied its properties only in the case of symmetric compatibilities. Both schemes are considerably simpler than Hummel and Zucker's continuous-time model [9] which requires a complicated projection operator. Moreover, the first scheme has no normalization phase, and this makes it particularly attractive for practical hardware implementations. Since our models automatically satisfy the constraints imposed by the structure of the labeling problem, they are also much simpler than Yu and Tsai's [8] and Li et al.'s [7] schemes, which have to take constraints into account either in the form of penalty functions or Lagrange multipliers.

The principal objective of this study is to analyze the dynamics of these relaxation schemes and to relate them to the classical theory of consistency developed by Hummel and Zucker [9]. We show that all the dynamical properties enjoyed by standard relaxation labeling algorithms do hold for ours. In particular, we prove that, when symmetric compatibility coefficients are employed, the models have a Liapunov function which rules their dynamical behavior, and this turns out to be (the negative of) a well-known consistency measure. Moreover, and most importantly, we prove that the fundamental dynamical properties of the systems are retained when the symmetry restriction is relaxed. We also study the properties of a simple discretization of the proposed models, which is useful in digital computer implementations. Some simulation results are presented which show how the models behave in practice and confirm their validity.

The outline of the paper is as follows. In Section 2, we briefly review Hummel and Zucker's consistency theory, which is instrumental for the subsequent development. In Section 3 we introduce the models and in Section 4 we present the main theoretical results, first for the symmetric and then for the non-symmetric case. Section 5 describes two ways of discretizing the models, and proves some results. In Section 6 we present our simulation results, and Section 7 concludes the paper.

2. Consistency and its properties

The labeling problem involves a set of objects $B = \{b_1, \ldots, b_n\}$ and a set of possible labels $\Lambda = \{1, \ldots, m\}$. The purpose is to label each object of B with one label of Λ. To accomplish this, two sources of information are exploited. The first one relies on *local* measurements which capture the salient features of each object viewed in isolation; classical pattern recognition techniques can be practically employed to carry out this task. The second source of information, instead, accounts for possible interactions among nearby labels and, in fact, incorporates all the contextual knowledge about the problem at hand. This is quantitatively expressed by means of a real-valued four-dimensional matrix of compatibility coefficients $R = \{r_{ij}(\lambda, \mu)\}$. The coefficient $r_{ij}(\lambda, \mu)$ measures the strength of compatibility between the hypotheses "b_i has label λ" and "b_j has label μ": high values correspond to compatibility and low values correspond to incompatibility. In our discussion, the compatibilities are assumed to be non-negative, i.e., $r_{ij}(\lambda, \mu) \geqslant 0$ for all $i, j = 1 \ldots n$ and $\lambda, \mu \in \Lambda$; as shown below, this is not a severe limitation. In this paper, moreover, we will not be concerned with the crucial problem of how to derive the compatibility coefficients. Suffice it to say that they can be either determined on the basis of statistical grounds [11,12] or, according to a more recent standpoint, adaptively learned over a sample of training data [13,14].

The initial local measurements are assumed to provide, for each object $b_i \in B$, an m-dimensional vector $\boldsymbol{p}_i^0 = (p_i^0(1), \ldots, p_i^0(m))^{\mathrm{T}}$ (where "T" denotes the usual transpose operation), such that $p_i^0(\lambda) \geqslant 0, i = 1 \ldots n, \lambda \in \Lambda$, and $\sum_\lambda p_i^0(\lambda) = 1, i = 1 \ldots n$. Each $p_i^0(\lambda)$ can be regarded as the initial, non-contextual degree of confidence of the hypothesis "b_i is labeled with label λ". By simply concatenating $\boldsymbol{p}_1^0, \boldsymbol{p}_2^0, \ldots, \boldsymbol{p}_n^0$ we obtain a weighted labeling assignment for the objects of B that will be denoted by $\boldsymbol{p}^0 \in \mathbb{R}^{nm}$. A relaxation labeling process takes as input the initial labeling assignment $\boldsymbol{p}^0$ and iteratively updates it taking into account the compatibility model R.

At this point, we introduce the space of weighted labeling assignments:

$$\mathbb{K}_{n,m} = \left\{ \boldsymbol{p} \in \mathbb{R}^{nm} \,\middle|\, p_i(\lambda) \geqslant 0, i = 1 \ldots n, \lambda \in \Lambda \text{ and} \right.$$

$$\left. \sum_{\lambda=1}^m p_i(\lambda) = 1, i = 1 \ldots n \right\}$$

which is a linear convex set of $\mathbb{R}^{nm}$. Every vertex of $\mathbb{K}_{n,m}$ represents an *unambiguous* labeling assignment, that is one which assigns exactly one label to each object. The set of these labelings will be denoted by $\mathbb{K}_{n,m}^*$:

$$\mathbb{K}_{n,m}^* = \left\{ \boldsymbol{p} \in \mathbb{K}_{n,m} \mid p_i(\lambda) = 0 \text{ or } 1, i = 1 \ldots n, \lambda \in \Lambda \right\}.$$

Moreover, a labeling p in the interior of $\mathbb{K}_{n,m}$ (i.e., $0 < p_i(\lambda) < 1$, for all i and λ) will be called *strictly ambiguous*.

Now, let $p \in \mathbb{K}_{n,m}$ be any labeling assignment. To develop a relaxation algorithm that updates p in accordance with the compatibility model, we need to define, for each object $b_i \in B$ and each label $\lambda \in \Lambda$, what is called a *support* function. This should quantify the degree of agreement between the hypothesis that b_i is labeled with λ, whose confidence is expressed by $p_i(\lambda)$, and the context. This measure is commonly defined as follows:

$$q_i(\lambda; p) = \sum_{j=1}^{n} \sum_{\mu=1}^{m} r_{ij}(\lambda, \mu) p_j(\mu). \tag{1}$$

Putting together the instances $q_i(\lambda; p)$, for all the $p_i(\lambda)$, we obtain an nm-dimensional support vector that will be denoted by $q(p)$.[1]

The following updating rule

$$p_i^{t+1}(\lambda) = \frac{p_i^t(\lambda)\, q_i^t(\lambda)}{\sum_\mu p_i^t(\mu)\, q_i^t(\mu)}, \tag{2}$$

where $t = 0, 1, \ldots$ denotes (discrete) time, defines the original relaxation labeling operator of Rosenfeld et al. [11], whose dynamical properties have recently been clarified [6]. In the following discussion we shall refer to it as the "classical" relaxation scheme.

We now briefly review Hummel and Zucker's theory of constraint satisfaction [9] which commences by providing a general definition of consistency. By analogy with the unambiguous case, which is more easily understood, a weighted labeling assignment $p \in \mathbb{K}_{n,m}$ is said to be *consistent* if

$$\sum_{\lambda=1}^{m} p_i(\lambda)\, q_i(\lambda; p) \geqslant \sum_{\lambda=1}^{m} v_i(\lambda)\, q_i(\lambda; p), \; i = 1 \ldots n \tag{3}$$

for all $v \in \mathbb{K}_{n,m}$. Furthermore, if strict inequalities hold in (3), for all $v \neq p$, then p is said to be *strictly consistent*. It can be seen that a necessary condition for p to be strictly consistent is that it is an unambiguous one, that is $p \in \mathbb{K}_{n,m}^*$.

In [9], Hummel and Zucker introduced the *average local consistency*, defined as

$$A(p) = \sum_{i=1}^{n} \sum_{\lambda=1}^{m} p_i(\lambda)\, q_i(\lambda) \tag{4}$$

and proved that when the compatibility matrix R is symmetric, i.e., $r_{ij}(\lambda, \mu) = r_{ji}(\mu, \lambda)$ for all i, j, λ, μ, then any local maximum $p \in \mathbb{K}_{n,m}$ of A is consistent. Basically, this follows immediately from the fact that, when R is symmetric, we have $\nabla A(p) = 2q$, $\nabla A(p)$ being the gradient of

---

A at p. Note that, in general, the converse need not be true since, to prove this, second-order derivative information would be required. However, by demanding that p be strictly consistent, this does happen [6].

Note that the concept of consistency is invariant under certain linear transformations of the compatibility matrix. In fact, let R be a compatibility matrix and let $\mathscr{C}(R)$ denote the set of consistent labelings with respect to R. From [9] we know that $\mathscr{C}(R) \neq \emptyset$. Let α and β be arbitrary constants, with $\alpha > 0$, and construct the matrix R' as follows: $r'_{ij}(\lambda, \mu) = \alpha r_{ij}(\lambda, \mu) + \beta$. Then, because $q'_i(\lambda) \equiv \sum_j \sum_\mu r'_{ij}(\lambda, \mu) p_j(\mu) = \alpha q_i(\lambda) + n\beta$, we have $\mathscr{C}(R) = \mathscr{C}(R')$. This justifies therefore our restriction to non-negative compatibilities.

3. Continuous-time relaxation labeling processes

The two relaxation labeling models studied in this paper are defined by the following systems of coupled differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t} p_i(\lambda) = p_i(\lambda) \left(q_i(\lambda) - \sum_\mu p_i(\mu)\, q_i(\mu) \right) \tag{5}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}t} p_i(\lambda) = p_i(\lambda) \frac{q_i(\lambda) - \sum_\mu p_i(\mu)\, q_i(\mu)}{\sum_\mu p_i(\mu)\, q_i(\mu)}. \tag{6}$$

For the purpose of the present discussion, $q_i(\lambda)$ denotes the linear support as defined in Eq. (1). As a matter of fact, many of the results proved below do not depend on this particular choice. More generally, the only requirements are that the support function be non-negative and, to be able to grant the existence and uniqueness of the solution of the differential equations, that it be of class C^1 [15].

In the first model we note that, although there is no explicit normalization process in the updating rule, the assignment space $\mathbb{K}_{n,m}$ is invariant under dynamics (5). This means that any trajectory starting in $\mathbb{K}_{n,m}$ will remain in $\mathbb{K}_{n,m}$. To see this, simply note that

$$\sum_\lambda \frac{\mathrm{d}}{\mathrm{d}t} p_i(\lambda) = \sum_\lambda p_i(\lambda) \left(q_i(\lambda) - \sum_\mu p_i(\mu)\, q_i(\mu) \right) = 0$$

which means that the interior of $\mathbb{K}_{n,m}$ is invariant. The additional observation that the boundary too is invariant completes the proof. The same result can be proven for the other model as well, following basically the same steps. The lack of normalization makes the first model, which we call the *standard* model, more attractive than Hummel and Zucker's projection-based scheme [9], since it makes it more amenable to hardware implementations and more acceptable biologically. The interest in the other model, called the *normalized* model and also studied by Stoddart [5], derives from the fact that, in

a way, it is the continuous-time translation of the classical Rosenfeld–Hummel–Zucker relaxation scheme [11] (see Section 5). We note that, using a linear support function (1), the dynamics of the models is invariant under a rescaling of the compatibility coefficients $r_{ij}(\lambda, \mu)$ as described at the end of Section 2. That is, if we define a set of new compatibility coefficients $r'_{ij}(\lambda, \mu) = \alpha r_{ij}(\lambda, \mu) + \beta$, with $\alpha > 0$ and $\beta \geq 0$, the orbit followed by the model remains the same, while the speed at which the dynamics evolve changes by a factor α.

As stated in the Introduction, one attractive feature of continuous-time systems is that they are readily mapped onto hardware circuitry. Figs. 1 and 2 show a circuit implementation for the standard and the normalized models, respectively. As expected, the standard model leads to a more economic implementation.

The fixed (or equilibrium) points of our dynamical systems are characterized by $(\mathrm{d}/\mathrm{d}t)\, \boldsymbol{p} = 0$ or, more explicitly, by $p_i(\lambda)\,[q_i(\lambda) - \sum_\mu p_i(\mu)\, q_i(\mu)] = 0$ for all $i = 1 \ldots n, \lambda \in \Lambda$. This leads us to the condition

$$p_i(\lambda) > 0 \Rightarrow q_i(\lambda) = \sum_\mu p_i(\mu)\, q_i(\mu) \tag{7}$$
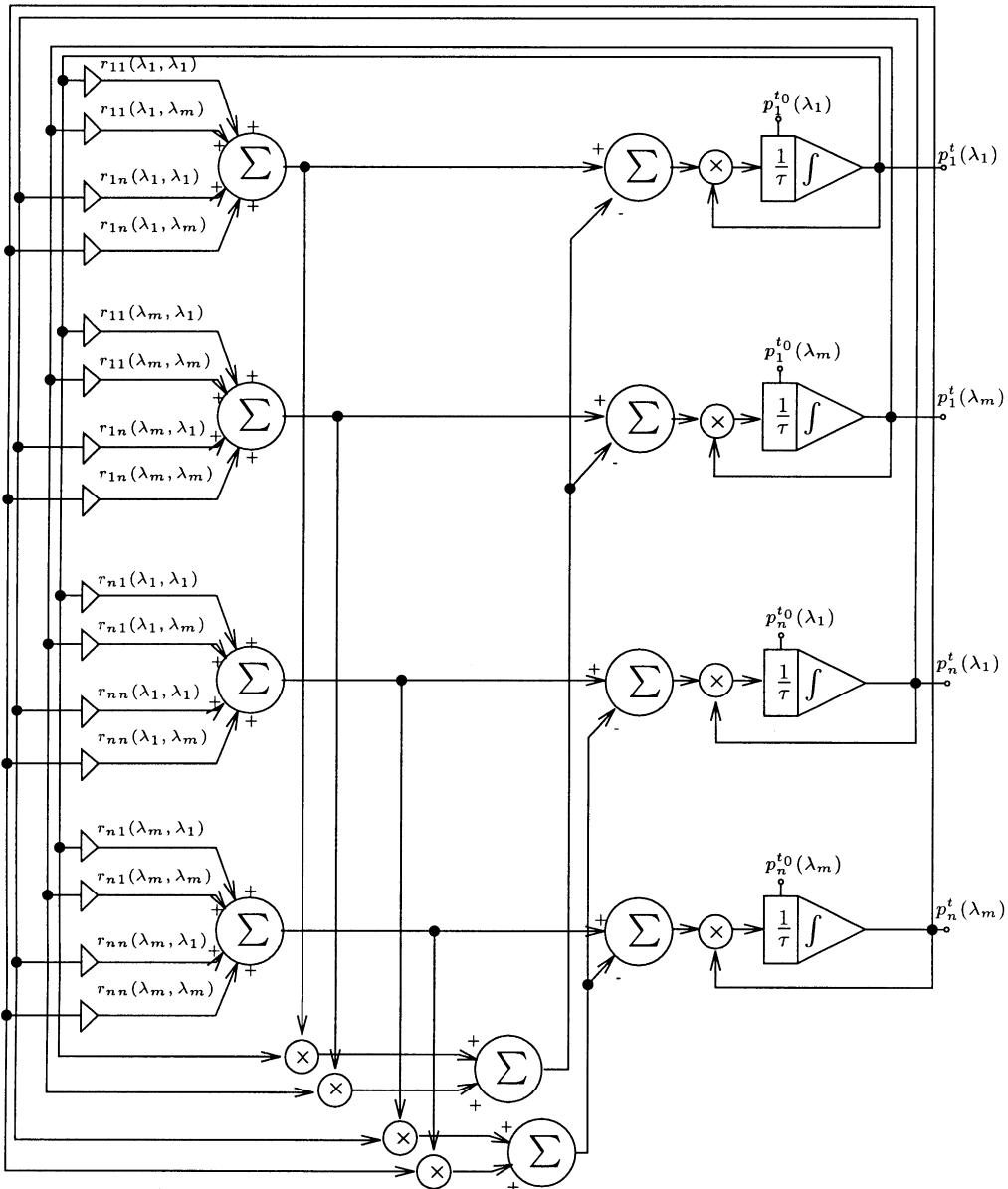
Fig. 1. Scheme of a circuit implementation for the standard model (5) — see, e.g., Ref. [3] for a description of the symbols adopted.
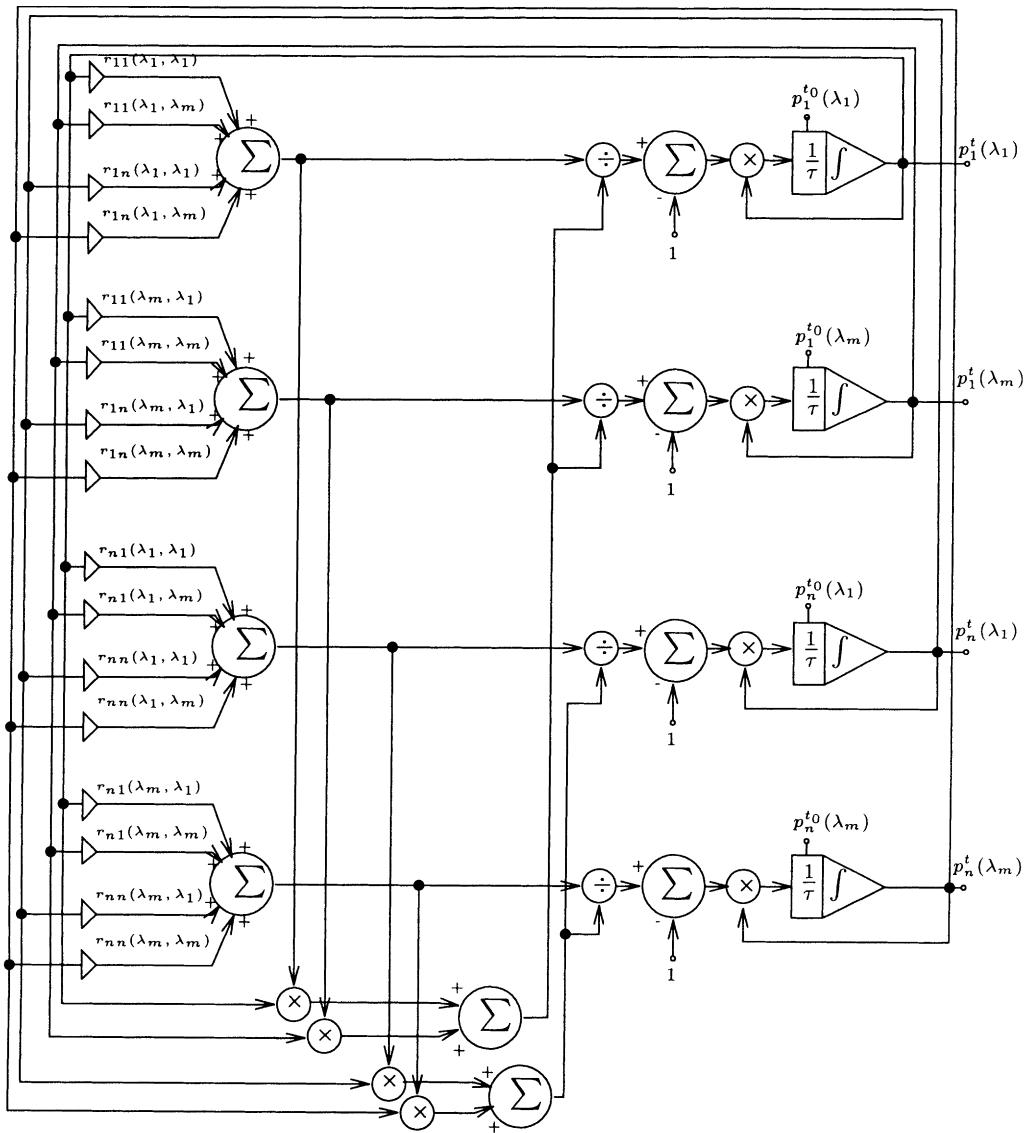
Fig. 2. Scheme of a circuit implementation for the normalized model (6) — see, e.g., Ref. [3] for a description of the symbols adopted.

which is the same condition we have for the Rosenfeld–Hummel–Zucker and Hummel–Zucker models.

The next result follows immediately from a characterization of consistent labelings proved in Ref. [6, Theorem 3.1].

Proposition 3.1. *Let $p \in \mathbb{K}_{n,m}$ be consistent. Then p is an equilibrium point for the relaxation dynamics* (5) *and* (6). *Moreover, if p is strictly ambiguous the converse also holds.*

This establishes a first connection between our continuous-time relaxation labeling processes and Hummel and Zucker's theory of consistency.

4. The dynamical properties of the models

In this section we study the dynamical properties of the proposed dynamical systems. Specifically, we show how our continuous-time relaxation schemes are intimately related to Hummel and Zucker's theory of consistency, and enjoy all the dynamical properties which hold for the classical discrete-time scheme (2), and Hummel and Zucker's projection-based model.

Before going into the technical details, we briefly review some instrumental concepts in dynamical systems theory; see Ref. [15] for details. Given a dynamical system, an equilibrium point x is said to be *stable* if,

whenever started sufficiently close to x, the system will remain near to x for all future times. A stronger property, which is even more desirable, is that the equilibrium point x be *asymptotically stable*, meaning that x is stable and in addition is a *local attractor*, i.e., when initiated close to x, the system tends towards x as time increases. One of the most fundamental tools for establishing the stability of a given equilibrium point is known as the Liapunov's direct method. It involves seeking a so-called *Liapunov* function, i.e., a continuous real-valued function defined in state space which is non-increasing along a trajectory.

4.1. Symmetric compatibilities

We present here some results which hold when the compatibility matrix R is symmetric, i.e., $r_{ij}(\lambda, \mu) = r_{ji}(\mu, \lambda)$, for all $i, j = 1 \ldots n$ and $\lambda, \mu \in \Lambda$. The following instrumental lemma, however, holds for the more general case of asymmetric matrices.

Lemma 4.1. *For all $p \in \mathbb{K}_{n,m}$ we have*

$$q(p) \cdot \frac{\mathrm{d}}{\mathrm{d}t} p \geqslant 0,$$

where "$\cdot$" represents the inner product operator, for both the standard and normalized relaxation schemes (5) *and* (6).

Proof. Let p be an arbitrary labeling assignment in $\mathbb{K}_{n,m}$. For the standard model we have

$$q(p) \cdot \frac{\mathrm{d}}{\mathrm{d}t} p = \sum_{i,\lambda} q_i(\lambda) p_i(\lambda) \left(q_i(\lambda) - \sum_{\mu} p_i(\mu) q_i(\mu) \right)$$

$$= \sum_i \left[\sum_\lambda p_i(\lambda) q_i^2(\lambda) - \left(\sum_\lambda p_i(\lambda) q_i(\lambda) \right)^2 \right].$$

Using the Cauchy–Schwartz inequality we obtain, for all $i = 1 \ldots n$,

$$\left(\sum_\lambda p_i(\lambda) q_i(\lambda) \right)^2 = \left(\sum_\lambda \sqrt{p_i(\lambda)} \cdot \sqrt{p_i(\lambda) q_i^2(\lambda)} \right)^2$$

$$\leqslant \sum_\lambda p_i(\lambda) \cdot \sum_\lambda p_i(\lambda) q_i^2(\lambda) = \sum_\lambda p_i(\lambda) q_i^2(\lambda).$$

Hence, since $\sum_\lambda p_i(\lambda) q_i^2(\lambda) \leqslant (\sum_\lambda p_i(\lambda) q_i(\lambda))^2$, we have $q(p) \cdot (\mathrm{d}/\mathrm{d}t) p \geqslant 0$.

The proof for the normalized model is identical; we just observe that

$$q(p) \cdot \frac{\mathrm{d}}{\mathrm{d}t} p = \sum_i \frac{\sum_\lambda q_i(\lambda) p_i(\lambda) \left(q_i(\lambda) - \sum_\mu p_i(\mu) q_i(\mu) \right)}{\sum_\mu p_i(\mu) q_i(\mu)}. \quad \square$$

A straightforward consequence of the previous lemma is the following important result, which states that, in the symmetric case, the average local consistency is always non-decreasing along the trajectories of our dynamical systems.

Theorem 4.2. *If the compatibility matrix R is symmetric, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t} A(p) \geqslant 0$$

for all $p \in \mathbb{K}_{n,m}$. In other words, $-A$ is a Liapunov function for the relaxation models (5) *and* (6).

Proof. Assuming $r_{ij}(\lambda, \mu) = r_{ji}(\mu, \lambda)$, we have

$$\frac{\mathrm{d}}{\mathrm{d}t} A(p) = 2 \sum_{i\lambda} \sum_{j\mu} r_{ij}(\lambda, \mu) p_j(\mu) \frac{\mathrm{d}}{\mathrm{d}t} p_i(\lambda)$$

$$= 2 q(p) \cdot \frac{\mathrm{d}}{\mathrm{d}t} p \geqslant 0. \quad \square$$

As far as the normalized scheme is concerned, this result has been proven by Stoddart [5]. By combining the previous result with the fact that strictly consistent labelings are local maxima of the average local consistency (see Ref. [6, Proposition, 3.4]) we readily obtain the following proposition.

Theorem 4.3. *Let p be a strictly consistent labeling and suppose that the compatibility matrix R is symmetric. Then p is an asymptotically stable stationary point for the relaxation labeling processes* (5) *and* (6) *and, consequently, is a local attractor.*

Therefore, in the symmetric case our continuous-time processes have exactly the same dynamical properties as the classical Rosenfeld–Hummel–Zucker model [6] and the Hummel–Zucker projection-based scheme [9].

4.2. Arbitrary compatibilities

In the preceding subsection we have restricted ourselves to the case of symmetric compatibility coefficients and have shown how, under this circumstance, the proposed continuous-time relaxation schemes are closely related to the theory of consistency of Hummel and Zucker. However, although symmetric compatibilities can easily be derived and asymmetric matrices can always be made symmetrical (i.e., by considering $R + R^{\mathrm{T}}$), it would be desirable for a relaxation process to work also when no restriction on the compatibility matrix is imposed [9]. This is especially true when the relaxation algorithm is viewed as a plausible model of how biological systems perform visual computation [16].

We now show that the proposed relaxation dynamical systems still perform useful computations in this case,

and their connection with the theory of consistency continues to hold. The main result is the following:

Theorem 4.4. *Let $p \in \mathbb{K}_{n,m}$ be a strictly consistent labeling. Then p is an asymptotically stable equilibrium point for the continuous-time relaxation labeling schemes defined in Eqs. (5) and (6).*

Proof. The first step in proving the theorem is to rewrite the models in the following way:

$$\frac{\mathrm{d}}{dt} p = F(p),$$

where, for all $i = 1 \ldots n$ and $\lambda \in \Lambda$,

$$F_i(\lambda)(p) = p_i(\lambda)\left(q_i(\lambda) - \sum_\mu p_i(\mu)q_i(\mu)\right)$$

for the standard model, and

$$F_i(\lambda)(p) = p_i(\lambda)\left(\frac{q_i(\lambda)}{\sum_\mu p_i(\mu)q_i(\mu)} - 1\right)$$

for the normalized model.

Let $DF(p)$ be the differential of F in p. We will show that if p is strictly consistent all eigenvalues of $DF(p)$ are real and negative. This means that p is a *sink* for the dynamical system and therefore an asymptotically stable point [15].

We begin by recalling that a strictly consistent labeling is necessarily non-ambiguous. Denoting by $\lambda(i)$ the unique label assigned to object b_i, we have

$$p_i(\lambda) = \begin{cases} 0 & \text{if } \lambda \neq \lambda(i) \\ 1 & \text{if } \lambda = \lambda(i) \end{cases} = \delta_{\lambda\lambda(i)}$$

where δ is the Kronecker delta, i.e., $\delta_{xy} = 1$ if $x = y$, and $\delta_{xy} = 0$ otherwise. Furthermore, from Eq. (3), we have $q_i(\lambda(i)) > q_i(\lambda)$ for all $\lambda \neq \lambda(i)$.

We first prove the theorem for the standard model. Deriving F with respect to $p_j(\rho)$, we have

$$\frac{\partial F_i(\lambda)}{\partial p_j(\rho)}(p) = \delta_{ij}\delta_{\lambda\rho}\left(q_i(\lambda) - \sum_\mu p_i(\mu)q_i(\mu)\right)$$
$$+ p_i(\lambda)\left(\frac{\partial q_i(\lambda)}{\partial p_j(\rho)} - \delta_{ij}q_i(\rho) - \sum_\mu p_i(\mu)\frac{\partial q_i(\mu)}{\partial p_j(\rho)}\right). \quad (8)$$

If we arrange the assignment vector in the following way:

$$p = (p_1(\lambda_1), \ldots, p_1(\lambda_m), \ldots, p_n(\lambda_1), \ldots, p_n(\lambda_m))^{\mathrm{T}}$$

and define the matrices $C_{ij} = (C_{ij}(\lambda, \mu))_{\lambda,\mu}$ as $C_{ij}(\lambda, \mu) = \partial F_i(\lambda)/\partial p_j(\mu)$, the differential takes the form

$$DF = \begin{vmatrix} C_{11} & C_{12} & \ldots & C_{1n} \\ C_{21} & C_{22} & \ldots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \ldots & C_{nn} \end{vmatrix}.$$

We can show that, if p is strictly consistent, $C_{ij} = 0$ if $i \neq j$. In fact, we have

$$\frac{\partial F_i(\lambda)}{\partial p_j(\rho)}(p) = p_i(\lambda)\left(\frac{\partial q_i(\lambda)}{\partial p_j(\rho)} - \sum_\mu p_i(\mu)\frac{\partial q_i(\mu)}{\partial p_j(\rho)}\right)$$

$$= \delta_{\lambda\lambda(i)}\left(\frac{\partial q_i(\lambda)}{\partial p_j(\rho)} - \frac{\partial q_i(\lambda(i))}{\partial p_j(\rho)}\right) = 0.$$

In this case the differential takes the form

$$DF = \begin{vmatrix} C_{11} & & 0 \\ & \ddots & \\ 0 & & C_{nn} \end{vmatrix}.$$

Analyzing the matrices C_{ii} we can see that these too take a particular form on strictly consistent assignments. In fact we have

$$\frac{\partial F_i(\lambda)}{\partial p_i(\rho)}(p) = \delta_{\lambda\rho}\left(q_i(\lambda) - \sum_\mu p_i(\mu)q_i(\mu)\right)$$

$$+ p_i(\lambda)\left(\frac{\partial q_i(\lambda)}{\partial p_i(\rho)} - q_i(\rho) - \sum_\mu p_i(\mu)\frac{\partial q_i(\mu)}{\partial p_i(\rho)}\right)$$

$$= \delta_{\lambda\rho}(q_i(\lambda) - q_i(\lambda(i)))$$

$$+ \delta_{\lambda\lambda(i)}\left(\frac{\partial q_i(\lambda)}{\partial p_i(\rho)} - q_i(\rho) - \frac{\partial q_i(\lambda(i))}{\partial p_i(\rho)}\right)$$

$$= \delta_{\lambda\rho}(q_i(\lambda) - q_i(\lambda(i))) - \delta_{\lambda\lambda(i)}q_i(\rho).$$

As we can notice, the non-zero values of C_{ii} are on the main diagonal and on the row $C_{ii}(\lambda, \rho)$ with $\lambda = \lambda(i)$. Thus, the eigenvalues of C_{ii} are the elements on the main diagonal. These are

$$q_i(\lambda) - q_i(\lambda(i)) \quad \text{for } \lambda \neq \lambda(i),$$
$$- q_i(\lambda(i)) \qquad \text{otherwise.} \quad (9)$$

Since p is strictly consistent, $q_i(\lambda) < q_i(\lambda(i))$ so all the eigenvalues are real and negative and not lower than $- q_i(\lambda(i))$. This tells us that the assignment is a sink, and hence an asymptotically stable point for the dynamical system.

We now prove the theorem for the normalized model. The fundamental steps to follow are the same as for the

standard model; we mainly have to derive the new values for the partial derivatives

$$\frac{\partial F_i(\lambda)}{\partial p_j(\rho)}(\boldsymbol{p})$$

$$= \frac{\delta_{ij}\delta_{\lambda\rho}q_i(\lambda) + p_i(\lambda)\,(\partial q_i(\lambda)/\partial p_j(\rho))}{\sum_\mu p_i(\mu)q_i(\mu)}$$

$$- \frac{p_i(\lambda)q_i(\lambda)\,(\delta_{ij}q_i(\rho) + \sum_\mu p_i(\mu)\,(\partial q_i(\mu)/\partial p_j(\rho)))}{(\sum_\mu p_i(\mu)q_i(\mu))^2} - \delta_{ij}\delta_{\lambda\rho}$$

$$= \delta_{ij}\delta_{\lambda\rho}\frac{q_i(\lambda)}{q_i(\lambda(i))} + \delta_{\lambda\lambda(i)}\frac{\partial q_i(\lambda)/\partial p_j(\rho)}{q_i(\lambda(i))} - \delta_{ij}\delta_{\lambda\lambda(i)}\frac{q_i(\lambda)q_i(\rho)}{q_i(\lambda(i))^2}$$

$$- \delta_{\lambda\lambda(i)}\frac{q_i(\lambda)\,(\partial q_i(\lambda(i))/\partial p_j(\rho))}{q_i(\lambda(i))^2} - \delta_{ij}\delta_{\lambda\rho}$$

$$= \delta_{ij}\delta_{\lambda\rho}\frac{q_i(\lambda)}{q_i(\lambda(i))} - \delta_{ij}\delta_{\lambda\lambda(i)}\frac{q_i(\rho)}{q_i(\lambda(i))} - \delta_{ij}\delta_{\lambda\rho}.$$

As the standard model, we have $C_{ij} = 0$ for $i \neq j$, and the matrices C_{ii} are non-zero only on the main diagonal and on the row related to the assignment $\lambda(i)$. Once more, then, the eigenvalues are equal to the elements on the main diagonal. These are

$$\frac{q_i(\lambda) - q_i(\lambda(i))}{q_i(\lambda(i))} \quad \text{for } \lambda \neq \lambda(i),$$

$$-1 \qquad\qquad \text{otherwise.} \tag{10}$$

Thus the eigenvalues are all real and negative and not lower than -1, i.e., strictly consistent assignments are sinks for system (6). □

The previous theorem is the analog to the fundamental local convergence result of Hummel and Zucker [9, Theorem 9.1], which is also valid for the classical relaxation scheme (2) [6, Theorem 6.4]. Note that, unlike Theorem 4.3, no restriction on the structure of the compatibility matrix is imposed here.

5. Discretizing the models

In order to simulate the behavior of the models on a digital computer, we need to make them evolve in discrete rather than continuous time steps. Two well-known techniques to approximate differential equations are the Euler method and the Runge–Kutta method. With the Euler method we have

$$p_i^{t+h}(\lambda) = p_i^t(\lambda) + hF_i^t(\lambda)(\boldsymbol{p}), \tag{11}$$

where h is the step size. This equation is advantageous since it can be computed in a very efficient way, so it is the ideal candidate for our simulations. We will prove that, given a certain integration step h, this model enjoys

all the dynamical properties shown for the continuous models it approximates.

In order to determine the difference in global behavior between the continuous models and the discrete approximations, we also use a finer discretization model: the IV grade Runge–Kutta method. This has been done on the assumption that this model would have a global dynamic behavior very similar to that of the continuous models. We have chosen the following Runge–Kutta scheme:

$$p_i^{t+h}(\lambda) = p_i^t(\lambda) + \frac{1}{6}k_1(i,\lambda) + \frac{2}{6}k_2(i,\lambda) + \frac{2}{6}k_3(i,\lambda)$$

$$+ \frac{1}{6}k_4(i,\lambda),$$

where the coefficients k_1, k_2, k_3, k_4 represent

$$k_1(i,\lambda) = hF_i(\lambda)(\boldsymbol{p}),$$

$$k_2(i,\lambda) = hF_i(\lambda)(\boldsymbol{p} + \tfrac{1}{2}\boldsymbol{k}_1),$$

$$k_3(i,\lambda) = hF_i(\lambda)(\boldsymbol{p} + \tfrac{1}{2}\boldsymbol{k}_2),$$

$$k_4(i,\lambda) = hF_i(\lambda)(\boldsymbol{p} + \boldsymbol{k}_3),$$

We will prove that the models discretized with Euler's method are well defined, that is, they map points in the assignment space $\mathbb{K}_{n,m}$ onto $\mathbb{K}_{n,m}$. Euler's scheme applied to our standard relaxation model (5) gives

$$p_i^{t+h}(\lambda) = p_i^t(\lambda) + hp_i^t(\lambda)\left(q_i^t(\lambda) - \sum_\mu p_i^t(\mu)q_i^t(\mu)\right).$$

We note that when h equals 1 the process is identical to the one recently proposed by Chen and Luh [17,18]. Their model imposes strict constraints on the compatibility coefficients to insure that $\mathbb{K}_{n,m}$ be invariant with respect to iterations of the process. However, it can be proven that, if an appropriate integration step h is chosen, it is not necessary to impose such constraints.

It is easy to prove that $\sum_\lambda p_i(\lambda)$ always equals 1:

$$\sum_\lambda p_i^{t+h}(\lambda) = 1 + h\left(\sum_\lambda p_i^t(\lambda)q_i^t(\lambda) - \sum_\lambda p_i^t(\lambda)\sum_\mu p_i^t(\mu)q_i^t(\mu)\right)$$

$$= 1.$$

But we have to prove that the iteration of the process never leads to negative assignments.

Proposition 5.1. *Let* $h \leqslant 1/q_i(\lambda; \boldsymbol{p})$ *for all* $i, \lambda, \boldsymbol{p}$. *Denoting by* E *the function generated applying Euler's scheme to model (5), then for all* $\boldsymbol{p} \in \mathbb{K}_{n,m}$, *we have* $E_i(\lambda)(\boldsymbol{p}) \geqslant 0$.

Proof. We have

$$p_i^{t+h}(\lambda) \geqslant p_i^t(\lambda) + hp_i(\lambda)\left(q_i^t(\lambda) - \sum_\mu p_i^t(\mu)\frac{1}{h}\right)$$

$$= p_i^t(\lambda) + hp_i^t(\lambda)\left(q_i^t(\lambda) - \frac{1}{h}\right)$$

$$\geqslant p_i^t(\lambda) - hp_i^t(\lambda)\frac{1}{h} = 0$$

which proves the proposition. □

If we use the linear support function (1), the integration step can be

$$h \leqslant \frac{1}{\max_{i\lambda} \{\sum_j \max_\mu r_{ij}(\lambda, \mu)\}}$$

It can readily be seen that this model also corrects deviation from the assignment space, provided that $p_i^t(\lambda) \geqslant 0$. In fact, given $\sum_\lambda p_i^t(\lambda) = 1 + \varepsilon$ we have

$$\sum_\lambda p_i^{t+h}(\lambda) = \sum_\lambda p_i^t(\lambda) + h \sum_\lambda p_i^t(\lambda) \left(q_i(\lambda) - \sum_\mu p_i^t(\mu) q_i^t(\mu) \right)$$

$$= (1 + \varepsilon) + h \left[\left(1 - \sum_\lambda p_i^t(\lambda) \right) \sum_\mu p_i^t(\mu) q_i^t(\mu) \right]$$

$$= 1 + \varepsilon - \varepsilon h \sum_\mu p_i^t(\mu) q_i^t(\mu).$$

As far as the normalized model is concerned, Euler's scheme yields

$$p_i^{t+h}(\lambda) = (1 - h) p_i^t(\lambda) + h \frac{p_i^t(\lambda) q_i^t(\lambda)}{\sum_\mu p_i^t(\mu) q_i^t(\mu)}.$$

As can easily be seen, with $h = 1$, this is the same equation that defines the classical model. Thus for $h = 1$ the model is well defined.

With an h lower than 1 the resulting assignment is a convex linear combination of p and the assignment resulting from applying one iteration of the classical method to p. Since the assignment space $\mathbb{K}_{n,m}$ is convex, the resulting assignment will also be in $\mathbb{K}_{n,m}$.

We can see that this model is also numerically stable. In fact, with $h = 1$, if we have $p_i(\lambda) \geqslant 0$, the model corrects any deviation from $\mathbb{K}_{n,m}$ in one step. On the other hand, with $h < 1$, if $\sum_\lambda p_i^t(\lambda) = 1 + \varepsilon$, we have

$$\sum_\lambda p_i^{t+h}(\lambda) = \sum_\lambda (1 - h) p_i^t(\lambda) + \sum_\lambda h \frac{p_i^t(\lambda) q_i^t(\lambda)}{\sum_\mu p_i^t(\mu) q_i^t(\mu)}$$

$$= (1 - h)(1 + \varepsilon) + h$$

$$= 1 + \varepsilon - h\varepsilon.$$

That is, the iteration of the model reduces the deviation from $\mathbb{K}_{n,m}$ at every step.

It is easy to prove that strictly consistent assignments are local attractors for these discrete models. In order to do this we must note that the differential of E is $I + hDF$; so, given an eigenvalue a of DF, there is an eigenvalue of DE equal to $1 + ha$. Furthermore, this property defines all eigenvalues of DE. As we have seen in Eq. (9), the eigenvalues of DF calculated for the standard model are all not lower than $\max_i \{- q_i(\lambda(i))\}$ and all strictly lower than 0; so, for any integration step lower than $1/q_i(\lambda)$ for all i and all λ, we have, for any eigenvalue b of DE, $b = 1 + ha \geqslant 1 - h(1/h) = 0$ and $b = 1 + ha < 1$. Thus, strictly consistent assignments are hyperbolic attractors for the system [19]. The eigenvalues of DF

calculated for the normalized model are all not lower than -1 and all strictly lower than 0 (10); so, for $h \leqslant 1$ the eigenvalues of DE are all not lower than 0 and all strictly lower than 1. Thus, in this case as well, strictly consistent assignments are hyperbolic attractors for the system.

6. Experimental results

In order to evaluate the practical behavior of the proposed models we conducted three series of experiments. Our goal was to verify that the models exhibit the same dynamical behavior as the classical relaxation scheme (2). The experiments were conducted using both the Euler and the Runge–Kutta discretizations described in the previous section. We did not considered the Euler discretization of the normalized model because, as seen before, it corresponds to the classical scheme, with a step size $h = 1$.

6.1. Labeling a triangle

The first set of simulations were conducted over the classical "triangle" problem introduced as a toy example in the seminal paper by Rosenfeld et al. [11]. The problem is to label the edges of a triangle as convex, concave, right- or left-occluding. Here, only eight possible labelings are possible (see Fig. 3 and Ref. [11] for details). The compatibility coefficients used were the same as those given in Ref. [11]. As a first test we verified whether the models' behaviors differ, starting from the eight initial assignments given Ref. [11]. From these starting points all the models gave the same sets of classifications. After this preliminary test, we generated 100 random assignments and used them as starting points for each model. The iterations were stopped when the sum of Kullback's I-directed divergence [20] between two successive assignments was lower than 10^{-7}. The average number of iterations that the models needed to reach the stopping criterion is shown in Table 1. All the models converged to a non-ambiguous assignment. Moreover, the Euler discretizations of our dynamics gave the same results as the classical model for all initial assignments, while the Runge–Kutta discretizations gave a different result only for one initial assignment. This single assignment was reached with the highest number of iterations of all the assignments generated. This is probably due to the symmetry of the problem: a similar problem can be seen with a uniform probability distribution among assignments. The iteration of each model should converge to the a priori probability of each classification, that is $\frac{3}{8}$ for each occluding edge and $\frac{1}{8}$ for convex or concave edges. What really happens is that the assignments start by heading towards the a priori distribution, but, after a few iterations, they head towards a non-ambiguous

The original image

A triangle over the background

A triangular hole on the background

Triangular concave folds

Triangular convex folds
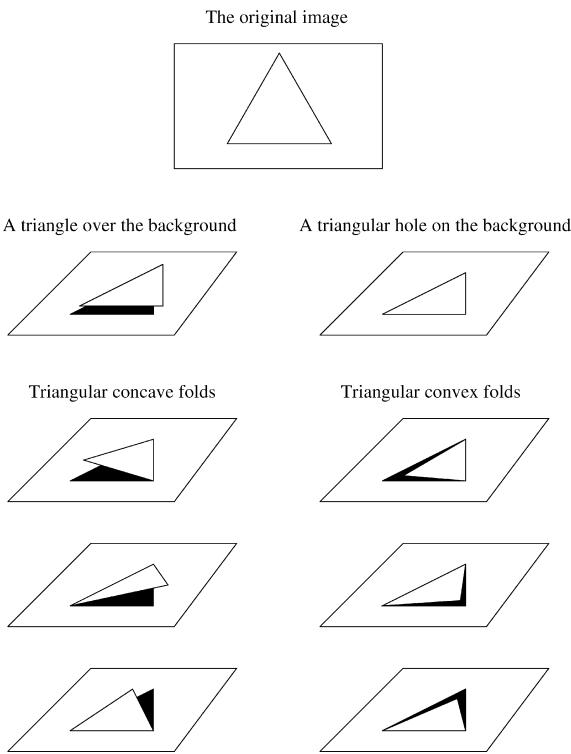
Fig. 3. A triangle and its possible interpretations.

Table 1
Average number of iterations for the triangle labeling problem

Model	Iterations
Classical (Eq. (2))	79.1
Standard, discretized with Euler's scheme	118.8
Normalized discretized with Runge–Kutta scheme	81.4
Standard, discretized with Runge–Kutta scheme	87.2

assignment. This happens because the a priori probability is not a hyperbolic attractor for the system. It is possible that a similar problem affected the only initial assignment that gave different results: the models headed toward different non-ambiguous solutions from a unique non-hyperbolic equilibrium that separates the orbits.

6.2. Finding maximal cliques in undirected graphs

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. Vertices i and j are called *adjacent* if they are connected by an edge. A *clique* of G is a subset of V such that every pair of vertices is adjacent. A clique C is called *maximal* if no strict superset of C is a clique, and the highest cardinality maximal clique is called a *maximum clique*. The cardinal-

ity of a set C will be denoted by $|C|$. The maximum clique problem (MCP) is to find a maximum clique in G, and it is known to be NP-hard even to approximate well (see Ref. [21] for a recent review).

The Motzkin–Straus theorem [22] allows us to formulate the MCP in terms of the following continuous quadratic optimization problem:

$$\text{maximize} \quad f(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{T}} A_G \boldsymbol{x}$$

$$\text{subject to} \quad \boldsymbol{x} \in \mathbb{K}_{1,n} \subseteq \mathbb{R}^n, \tag{12}$$

where n is the number of vertices in G, and $A_G = (a_{ij})$ is its *adjacency matrix*, i.e., the $n \times n$ matrix defined as $a_{ij} = 1$ if $(i, j) \in E$, and $a_{ij} = 0$ if $(i, j) \notin E$. Specifically, it can be proven that a subset C of vertices in G is a maximum clique if and only if its *characteristic vector* $\boldsymbol{x}^c \in \mathbb{K}_{1,n}$, defined as $x_i^c = 1/|C|$ if $i \in C$ and $x_i^c = 0$ otherwise, is a global solution of program (12). In this case, the size of the maximum clique C is related to the global maximum as follows: $f(\boldsymbol{x}^c) = 1 - 1/|C|$. It can also be seen that *local* solutions to Eq. (12) are in correspondence with *maximal* cliques of G [23].

Note that the function f of program (12) is exactly the average local consistency defined in Eq. (4) for a 1-object, n-label labeling problem, with the (symmetric) adjacency matrix A_G playing the role of the compatibility matrix R. In Ref. [24], this observation has motivated the use of the classical (discrete-time) Rosenfeld–Hummel–Zucker relaxation labeling algorithm (2) as an efficient heuristic to approximately solve the MCP. The algorithm, starting from any point in $\mathbb{K}_{1,n}$, iteratively maximizes the function f and eventually approaches a fixed point which corresponds to a local maximizer of f of $\mathbb{K}_{1,n}$. This solution is therefore employed to get an estimate of the size of the maximum clique in G.

As the second series of experiments for the work presented here, we tested our continuous-time dynamics on the MCP, following the approach just described. The models were applied on 10 randomly generated 100-vertex graphs with 90% connectivity. The algorithms were started from the barycenter of $\mathbb{K}_{1,n}$ as in Ref. [24], i.e. from the vector $(1/n, \ldots, 1/n)^{\mathrm{T}}$, and they were allowed to iterate until the Kullback's divergence between two successive vectors was lower than 10^{-20}. Table 2 shows the average number of iterations needed by the models to converge. As far as the long-term behavior of the models is concerned, a picture similar to the triangle experiments emerged. The Euler discretization of the standard model returned the same solutions as the classical scheme (2) in all the cases. The Runge–Kutta discretizations, instead, differed from the classical scheme in one case only, where they found a clique size greater than that found by the classical scheme (i.e, 28 instead of 27 for both the standard and the normalized models). These results are in accordance with the recent empirical findings of Bomze and Rendl [25].

Table 2
Average number of iterations for the maximal clique experiments

Model	Iterations
Classical (Eq. (2))	1688.9
Standard, discretized with Euler's scheme	1717.6
Normalized discretized with Runge–Kutta scheme	1747.8
Standard, discretized with Runge–Kutta scheme	1780.0

6.3. Random asymmetric compatibility matrices

The third set of simulations was carried out by generating random (asymmetric) compatibility matrices. Since in this case there is no Liapunov function which rules the dynamics of the models, this is the set of tests which most effectively points out differences in the process' behavior. Ten random compatibility matrices were generated for this experiment and for each matrix the various models were started from 10 random assignments. Hence we made a hundred tests for each model. The assignment space dimension was five objects ($n = 5$) and three labels ($m = 3$). The stopping criterion was the same as the triangle experiments. Table 3 reports the average number of iterations needed to reach the stopping criterion.

Since there is no underlying scheme on the pattern of compatibility coefficients, we do not expect the models to always converge to a non-ambiguous assignment. The aim of this set of tests was to verify whether, when the classical model converges to a non-ambiguous assignment, the other models converge to the same assignment too. In our experiments the classical model (2) converged to a non-ambiguous assignment 22 times out of 100 trials. The Runge–Kutta discretization of both models converged to the same assignments 20 times, while the Euler discretization of the standard model returned the same assignments 17 times. These differences are not surprising since Theorem 4.4 guarantees us that the models exhibit the same dynamical behavior only in the vicinity of consistent labelings, and in the asymmetric case no Liapunov function has been found.

Table 3
Average number of iterations for the random compatibility experiments

Model	Iterations
Classical (Eq. (2))	294.0
Standard, discretized with Euler's scheme	643.1
Normalized discretized with Runge–Kutta scheme	260.1
Standard, discretized with Runge-Kutta scheme	324.5

7. Conclusions

In this paper we have presented and analyzed two relaxation labeling processes. In contrast with the standard approach, these models evolve through continuous-time rather than discrete-time dynamics. This fact makes the study of their properties simpler and more elegant, permits the design of analog hardware implementations, and makes them more plausible biologically. We have studied the properties of the models and have shown that they are intimately related to Hummel and Zucker's classical theory of consistency. More precisely, we have proven that when the compatibility matrix is symmetric the models possess a quadratic Liapunov function, and (strictly) consistent labelings are asymptotically stable stationary points for them. In the more general case of asymmetric compatibilities, we have shown that strictly consistent labelings still are asymptotically stable for the proposed dynamical systems, although no Liapunov function has been found. These are exactly the same properties enjoyed by classical relaxation labeling schemes. The dynamics of the models discretized through Euler's scheme has also been studied. Experimental results on various test problems using the Euler as well as the Runge–Kutta discretizations have been presented which show the practical behavior of the proposed algorithms. It turned out that the models exhibit essentially the same global dynamics as the classical Rosenfeld–Hummel–Zucker scheme when symmetric compatibility matrices are used. In the case of asymmetric matrices, however, some differences emerge, which can be explained by the lack of a Liapunov function driving the model's behavior.

References

[1] D.H. Ballard, C.M. Brown, Computer Vision, Prentice-Hall, Englewood Cliffs, NJ, 1982.
[2] J. Kittler, J. Illingworth, Relaxation labeling algorithms — A review, Image Vision Comput. 3 (1985) 206–216.
[3] A. Cichoki, R. Unbehauen, Neural Networks for Optimization and Signal Processing, Wiley, New York, 1993.
[4] J.J. Hopfield, Neurons with graded response have collective computational properties like those of two-state neurons, Proceedings of the National Academy of Sciences USA 81 (1984) 3088–3092.

[5] A.J. Stoddart, M. Petrou, J. Kittler, On the foundations of probabilistic relaxation with product support, J. Math. Imaging Vision 9 (1998) 29–48.

[6] M. Pelillo, The dynamics of nonlinear relaxation labeling processes, J. Math. Imaging Vision 7 (4) (1997) 309–323.

[7] S.Z. Li, W.Y.C. Soh, E.K. Teoh, Relaxation labeling using augmented Lagrange–Hopfield method, Pattern Recognition 31 (1) (1998) 73–81.

[8] S.-S. Yu, W.-H. Tsai, Relaxation by the Hopfield neural network, Pattern Recognition 25 (2) (1992) 197–209.

[9] R.A. Hummel, S.W. Zucker, On the foundations of relaxation labeling processes, IEEE Trans. Pattern Anal. Mach. Intell. 5 (1983) 267–287.

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

[11] A. Rosenfeld, R.A. Hummel, S.W. Zucker, Scene labeling by relaxation operations, IEEE Trans. Systems Man Cybernet. 6 (6) (1976) 420–433.

[12] S. Peleg, A. Rosenfeld, Determining compatibility coefficients for curve enhancement relaxation processes, IEEE Trans. Systems Man Cybernet. 8 (1978) 548–555.

[13] M. Pelillo, M. Refice, Learning compatibility coefficients for relaxation labeling processes, IEEE Trans. Pattern Anal. Mach. Intell. 16 (9) (1994) 933–945.

[14] M. Pelillo, A.M. Fanelli, Autoassociative learning in relaxation labeling networks, Pattern Recognition Lett. 18 (1) (1997) 3–12.

[15] M.W. Hirsch, S. Smale, Differential Equations, Dynamical Systems, and Linear Algebra, Academic Press, New York, 1974.

[16] S.W. Zucker, A. Dobbins, L. Iverson, Two stages of curve detection suggest two styles of visual computation, Neural Comput. 1 (1989) 68–81.

[17] Q. Chen, J.Y.S. Luh, Ambiguity reduction by relaxation labeling, Pattern Recognition 27 (1) (1994) 165–180.

[18] Q. Chen, J.Y.S. Luh, Relaxation labeling algorithm for information integration and its convergence, Pattern Recognition 28 (11) (1995) 1705–1722.

[19] R.L. Devaney, An Introduction to Chaotic Dynamical Systems, Addison-Wesley, Reading, MA, 1989.

[20] S. Kullback, Information Theory and Statistics, Wiley, New York, 1959.

[21] I.M. Bomze, M. Budinich, P.M. Pardalos, M. Pelillo, The maximum clique problem, in: D.-Z. Du, P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Vol. 4, Kluwer Academic Publishers, Boston, MA, 1999.

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

[23] M. Pelillo, A. Jagota, Feasible and infeasible maxima in a quadratic program for maximum clique, J. Artif. Neural Networks 2 (4) (1995) 411–420.

[24] M. Pelillo, Relaxation labeling networks for the maximum clique problem, J. Artif. Neural Networks 2 (4) (1995) 313–328.

[25] I.M. Bomze, F. Rendl, Replicator dynamics for evolution towards the maximum clique: variations and experiments, in: R. De Leone, A. Murli, P.M. Pardalos, G. Toraldo (Eds.), High Performance Algorithms and Software in Nonlinear Optimization, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998, pp. 53–68.

About the Author—ANDREA TORSELLO was born on June 18, 1973 and received the "Laurea" degree with honors in Computer Science from the University of Venice, Italy, in 1997. Since 1998 he is with MicroStrategy, a leading worldwide provider of enterprise decision support system software. His research interests are in pattern recognition and computer vision.

About the Author—MARCELLO PELILLO was born in Taranto, Italy, on June 1, 1966. He received the "Laurea" degree with honors in Computer Science from the University of Bari, Italy, in 1989. From 1988 to 1989 he spent about a year at the IBM Scientific Center in Rome, where he was involved in studies on natural language and speech processing. In 1991 he joined the Department of Computer Science at the University of Bari, Italy, as Assistant Professor. Since 1995 he is with the Department of Computer Science at the University of Venice, Italy. He held visiting research positions at Yale University (USA), McGill University (Canada), the University of Vienna (Austria), and the University of York (UK). His research interests are in the areas of pattern recognition, computer vision, and neural networks, where he has published over 60 papers in refereed journals, handbooks, and conference proceedings. He has been the program co-chair of the First and the Second *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition* (EMMCVPR) held in Venice, Italy, in May 1997, and in York, UK, in July 1999, respectively. He has been a Guest Co-editor of a special issue of the journal *Pattern Recognition* devoted to this theme, and he is guest co-editing a special issue of *IEEE Transactions on Pattern Analysis and Machine Intelligence* devoted to "Graph algorithms in computer vision". He is on the Editorial Board of *Pattern Recognition*, and he is a member of the IEEE Computer Society and the Pattern Recognition Society.