A Class of Solvable Consistent Labeling Problems

Boris Flach[1] and Michail I. Schlesinger[2]

[1] Institute of Artificial Intelligence, Dresden University of Technology, D-01062 Dresden, Germany, Phone +49 351 4638268, Fax +49 351 4638369
bflach@ics.inf.tu-dresden.de

[2] International Centre of Information Technologies and Systems, National Academy of Science of Ukraine, prospect Academica Gluchkova, 40, 252022 Kiev 22, Ukraine
schles@image.kiev.ua

Abstract. The structural description ansatz often used for representing and recognizing complex objects leads to the consistent labeling problem or to some optimization problems on labeled graphs. Although this problems are NP-complete in general it is well known that they are easy solvable if the underlying graph is a tree or even a partial m-tree (i.e its treewidth is m). On the other hand the underlying graphs arising in image analysis are often lattices or even fully connected. In this paper we study a special class of consistent labeling problems where the label set is ordered and the predicates preserve some structure derived from this ordering. We show that consistent labeling can be solved in polynomial time in this case even for fully connected graphs. Then we generalize this result to the "MaxMin" problem on labeled graphs and show how to solve it if the similarity functions preserve the same structure.

1 Introduction

Structural description is one of the most general methods for representing and recognizing complex real world objects and thus very popular in image analysis. Especially attributed or labeled graphs are often used as an effective means of structural description: A complex object is composed of primitives which have to fulfil some neighbourhood constraints. The primitives are represented by labels attached to the vertices of a graph whereas the constraints can be thought as predicates on pairs of labels and are attached to the edges of the graph. Recognition of objects modelled in this way could be divided into two stages: First, some e.g. local features are measured in order to obtain the primitives located in image fragments corresponding to the graph nodes r. Of course we cannot expect to have unique answers by local measurements. So the answer will be for instance a subset of possible primitives or some similarity function for each vertex. In the first case the next stage of recognition is equivalent to a *consistent labeling problem*, in the second case – to some *optimization problem* on labeled graphs.

A well known and popular example of this kind of recognition are Hidden Markov Models used in speech recognition, where primitives are phonemes or

F.J. Ferri et al. (Eds.): SSPR&SPR 2000, LNCS 1876, pp. 462–471, 2000.

phoneme groups. Obviously almost all problems can be solved in linear time for these models because the underlying graph is a simple chain [8]. In contrast, the situation in image analysis is much harder: the graphs under consideration are often lattices or even fully connected. In general consistent labeling and most optimization problems on labeled graphs[1] are NP-complete. Therefore all known algorithms for solving these problems in the general case (i.e. without further assumptions) scale exponentially with the number of vertices of the graph [3].

The aforementioned problems are easy solvable if the underlying graph is a tree or even a partial m-tree (i.e. its treewidth is m): then algorithms of complexity $n\,|K|^{m+1}$ are known, where K is the set of used primitives and n is the number of vertices of the graph [6,1]. On the other hand these algorithms are not very useful in image analysis: an $n \times n$ rectangular lattice is a partial n-tree!

Despite the importance of the consistent labeling problem its long history lacks attempts to investigate how its complexity depends on used predicates i.e.: Do there exist classes of predicates, so that the aforementioned problems are solvable in polynomial time even for fully connected graphs?

In this paper we study a special class of consistent labeling problems where the label set K is ordered and the predicates preserve some structure derived from this ordering.[2] We show that consistent labeling can be solved in polynomial time in this case even for fully connected graphs. Then we generalize this result to the "MaxMin" problem on labeled graphs and show how to solve it if the similarity functions preserve the same structure.

2 Consistent Labeling and Related Optimization Problems

In this section we introduce a formal notion of the consistent labeling problem and some related optimization problems arising in image analysis for the sake of completeness and self consistence of this paper.

Let $\mathcal{G} = (R, E)$ be an undirected graph with vertices R and edges E. Let K be a finite set of labels (often called symbols in the context of structural recognition). A labeling or symbol field is a mapping $y\colon R \mapsto K$ assigning a symbol $y(r)$ to each vertex $r \in R$. The set of all symbol fields is denoted by $\mathcal{A}(R, K)$.

Neighbourhood constraints are represented by predicates $\chi_{ij}\colon K \times K \mapsto \{0, 1\}$ attached to edges $(r_i, r_j) \in E$ of the graph. These predicates define allowable pairs of symbols on the edges: Only symbol pairs (k_1, k_2) with $\chi_{ij}\,(k_1, k_2) = 1$ are allowed on edge $(r_i, r_j) \in E$. The field of all predicates $\chi_{ij}(.,.)$ attached to

[1] From our point of view the *stochastic relaxation labeling* introduced by Hummel, Rosenfeld and Zucker [4,5,2] does not represent a realistic optimization problem, because they redefine consistence by some particular kind of local stability after transition to real valued similarity functions.

[2] First results on a narrower class were reported in [7].

the edges $(r_i, r_j) \in E$ of the graph $\mathcal{G}$ is denoted by $\widehat{\chi}$ and called a *local conjunctive predicate* (LCP) on $\mathcal{G}$.

A symbol field $y \in \mathcal{A}(R, K)$ is a solution of $\widehat{\chi}$ if

$$\chi_{ij}[y(r_i),\, y(r_j)] = 1 \tag{1}$$

for each edge $(r_i, r_j) \in E$. It is of course very easy to verify whether y is a solution of $\widehat{\chi}$ by proving (1) for each edge of $\mathcal{G}$. In contrast the consistent labeling problem is much harder: We have to prove whether a given $\widehat{\chi}$ has solutions. A typical situation in image analysis is that given $\mathcal{G}$ and $\widehat{\chi}$ describing the model, local measurements "narrow" the set of possible symbols for each vertex: We obtain a subset $K_i \subset K$ for each $r_i \in R$ and have to prove whether $\widehat{\chi}$ has solutions y where $y(r_i) \in K_i$, $\forall r_i \in R$.

Often we are in a "weaker" situation where local measurements give similarity values for symbols or symbol pairs, expressing fuzzy subsets rather than sharp subsets. In this case the predicates χ_{ij} are replaced by real valued functions $f_{ij} \colon K \times K \mapsto \mathbb{R}$ and some optimization problem is to be solved. Often it is the MaxMin problem:

$$\max_{y \in \mathcal{A}(R,K)} \min_{(r_i, r_j)} f_{ij}[y(r_i), y(r_j)] \tag{2}$$

i.e. find the symbol field where the smallest similarity on some edge is as big as possible. Another typical problem is to find the symbol field with the highest sum of similarities:

$$\max_{y \in \mathcal{A}(R,K)} \sum_{(r_i, r_j)} f_{ij}[y(r_i), y(r_j)] \ . \tag{3}$$

3 Closed Predicates on Ordered Sets

In this section we introduce the class of closed predicates which then will be used to form a class of consistent labeling problems. We split the definition of these predicates into two parts: First, we endow the symbol set K with an additional structure. Closed predicates are then those which preserve this structure.

Suppose the set K of symbols is *ordered*. Let $\mathcal{U}$ be the system of all *intervals* of K i.e. subsets of the type $\{k \in K \mid k_1 \leqslant k \leqslant k_2\}$. Then $\mathcal{U}$ meets the definition of a *hull system* as used in universal algebra, because

1. K is an interval, i.e. $K \in \mathcal{U}$.
2. $\mathcal{U}$ is closed under intersections, i.e. if $U, U' \in \mathcal{U}$ then $U \cap U' \in \mathcal{U}$.

Usually subsets which are elements of a hull system, are called hulls or closed subsets. The *closure* of an arbitrary subset A is defined as the smallest closed subset containing A:

$$\mathrm{Cl}(A) = \bigcap \{U \in \mathcal{U} \mid A \subset U\}$$

Examples of other hull systems are closed sets of a topological space or convex sets of a vector space. Our special hull system has another striking property:

3. Whenever $\mathcal{B} \subset \mathcal{U}$ is a set of intervals with pairwise nonempty intersections i.e. $U \cap U' \neq \emptyset, \forall U, U' \in \mathcal{B}$, then their intersection is not empty: $\bigcap_{U \in \mathcal{B}} U \neq \emptyset$.

The structure of this hull system is inherited to every subset of K: Let $K_1 \subset K$ be an arbitrary subset, then

$$\mathcal{U}_{K_1} = \{U' \subset K_1 \mid \exists\, U \in \mathcal{U} \colon U' = U \cap K_1\}$$

is a hull system fulfilling 1–3.

Let $\chi \colon K \times K \mapsto \{0,1\}$ be a predicate on $K \times K$. We interprete χ equivalently as a *subset* $\chi \subset K \times K$: $\{(k_1, k_2) \mid \chi(k_1, k_2) = 1\}$. A third equivalent meaning is to interprete χ as a *relation* on K. Therefore the set of all predicates on $K \times K$ can be endowed with the operations $\chi \cap \chi'$ and $\chi \circ \chi'$, where

$$(\chi \circ \chi')\,(k_1, k_2) = 1 \Leftrightarrow \exists\, k_3 \colon \chi(k_1, k_3) = 1 \;\text{ and }\; \chi'(k_3, k_2) = 1$$

denotes multiplication of relations.

Now we are ready to define the class of closed predicates, i.e. predicates preserving the structure of the hull system $\mathcal{U}$. Let π_1 and π_2 denote the projections from $K \times K$ onto the first resp. second component: If $M \subset K \times K$, then

$$\pi_1(M) = \{k_1 \mid \exists k_2 \colon (k_1, k_2) \in M\} \text{ and } \pi_2(M) = \{k_2 \mid \exists k_1 \colon (k_1, k_2) \in M\}\,.$$

Let χ be a predicate on $K \times K$ considered as a subset with $K_1 = \pi_1(\chi)$ and $K_2 = \pi_2(\chi)$. Then χ induces the mappings $F_\chi \colon \mathcal{P}(K_1) \mapsto \mathcal{P}(K_2)$ and $F_\chi^{-1} \colon \mathcal{P}(K_2) \mapsto \mathcal{P}(K_1)$ where $\mathcal{P}(K)$ denotes the power set of K. They map subsets of $K_1 \subset K$ on subsets of $K_2 \subset K$ and vice versa. These mappings are defined as follows:

$$F_\chi(V_1) = \pi_2\left[\,(V_1 \times K_2) \cap \chi\,\right] \;\text{ and }\; F_\chi^{-1}(V_2) = \pi_1\left[\,(K_1 \times V_2) \cap \chi\,\right]$$

where $V_1 \subset K_1$ and $V_2 \subset K_2$ (see Fig. 1). Please remark that F_χ^{-1} means the inverse of F_χ only if it is invertible. Nevertheless $V_1 \subset F_\chi^{-1}(F_\chi(V_1))$ and $V_2 \subset F_\chi(F_\chi^{-1}(V_2))$ hold for every χ and every $V_1 \subset K_1$ and $V_2 \subset K_2$.

Definition 1. *Let χ be a predicate on $K \times K$ where K is ordered and $K_1 = \pi_1(\chi)$, $K_2 = \pi_2(\chi)$. Then χ is called closed if F_χ and F_χ^{-1} map hulls of $\mathcal{U}_{K_1}$ on hulls of $\mathcal{U}_{K_2}$ and vice versa.*

Example 1. Let $|K| = 4$. The predicate depicted left in Fig. 2 is closed, whereas that on right is not.

Lemma 1. *Let K be ordered and χ, χ' be closed predicates on $K \times K$. Then $\chi \cap \chi'$ and $\chi \circ \chi'$ are closed predicates.*

Proof. In order to simplify subsequent steps of the proof we remark that intersections of closed predicates with product subsets are again closed predicates: Let χ be a closed predicate on $K \times K$ and K_1, $K_2 \subset K$. Then the predicate $\chi \cap (K_1 \times K_2)$ is closed.

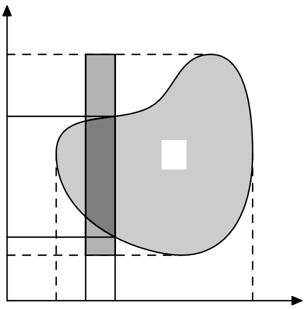

Fig. 1. Illustration of the mapping F_χ: The intersections of the dotted lines with the components depict the borders of $K_1 = \pi_1(\chi)$ and $K_2 = \pi_2(\chi)$. The vertical grey bar depicts $V_1 \times K_2$. Its intersection with χ projected on K_2 gives $F_\chi(V_1)$.

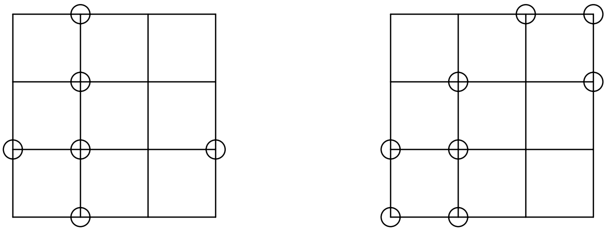

Fig. 2. Two predicates on $K \times K$ depicted as subsets. The left is closed, the right is not.

Let χ', χ'' be closed predicates on $K \times K$ and $\chi = \chi' \cap \chi''$. Without loss of generality we assume that $\pi_1(\chi) = \pi_2(\chi) = K$. Let F, F' and F'' denote the assotiated mappings from $\mathcal{P}(K)$ into $\mathcal{P}(K)$.

Let us assume now that χ is not closed. Then there must be an interval $A_1 \subset K$, so that $A_2 = F(A_1)$ is not closed. Let k be an element of $\mathrm{Cl}(A_2)$ not contained in A_2 i.e. $(A_1 \times k) \cap \chi = \emptyset$. We consider the sets

$$B_1 = {F'}^{-1}(k) = \pi_1\left[(K \times k) \cap \chi'\right] \text{ and } C_1 = {F''}^{-1}(k) = \pi_1\left[(K \times k) \cap \chi''\right] .$$

According to our assumptions it follows then, that

- B_1 and C_1 are nonempty and $B_1 \cap C_1 \neq \emptyset$.
- $\mathrm{Cl}(A_2) \subset F'(A_1)$ holds and therefore $B_1 \cap A_1 \neq \emptyset$. Similarly $C_1 \cap A_1 \neq \emptyset$.

Because A_1, B_1 and C_1 are closed (i.e. intervals) and hence $A_1 \cap B_1 \cap C_1 \neq \emptyset$, $k \in A_2$ follows on contradiction. Hence $\chi = \chi' \cap \chi''$ must be closed.

Let us prove now that $\chi = \chi' \circ \chi''$ is closed whenever χ' and χ'' are closed. Without loss of generality we assume that $\pi_1(\chi') = \pi_2(\chi'')$. But in this case

$$F(A) = F''\left(F'(A)\right) \text{ and } F^{-1}(A) = {F'}^{-1}\big({F''}^{-1}(A)\big)$$

and therefore χ is closed. $\square$

4 Consistent Labeling with Closed Predicates

In this section we consider LCP with closed local predicates. We will show that consistent labeling for such LCP can be solved in polynomial time without any restrictions on the underlying graph. We present a parallel algorithm that solves the problem and generates a weak description of the solution set. A sequential version of this algorithm is presented as well.

Theorem 1. *Let $\mathcal{G} = (R, E)$ be a fully connected graph with n vertices and K be an ordered symbol set. The consistent labeling problem for a LCP $\widehat{\chi}$ on $\mathcal{G}$ can be solved in polynomial time if $\widehat{\chi}$ is closed i.e. all local predicates of $\widehat{\chi}$ are closed.*

Proof. Consider the following iterative algorithm:[3]

$$\begin{aligned} \chi_{ij}^{(0)} &= \chi_{ij} \\ \chi_{ij}^{(t+1)} &= \chi_{ij}^{(t)} \cap \Big[\bigcap_{k \neq i,j} \big(\chi_{ik}^{(t)} \circ \chi_{kj}^{(t)} \big) \Big] \end{aligned} \tag{4}$$

The series of LCP $\widehat{\chi}^{(t)}$ reaches a fixpoint $\widehat{\chi}^{*}$ after at most $n^2 |K|^2 / 2$ iterations. This is because there are $n^2/2$ predicates each one with $|K|^2$ binary entries and after each iteration at least one of this $n^2|K|^2/2$ entries changes from 1 to 0 (They never change from 0 to 1).

The solution set of $\widehat{\chi}$ is nonempty if every χ_{ij}^{*} is nonempty in the fixpoint. If at least one χ_{ij}^{*} is empty, then $\widehat{\chi}$ has no solutions. This is true because each solution of $\widehat{\chi}^{(t)}$ is also a solution of $\widehat{\chi}^{(t+1)}$. Hence $\widehat{\chi}$ has solutions only if $\widehat{\chi}^{*}$ has solutions.

Let us consider the situation where all χ_{ij}^{*} are nonempty. Then all sets $\pi_1(\chi_{ij}^{*})$ are nonempty. Furthermore, these sets coincide for fixed i: Suppose there is a pair j, k for which $\pi_1(\chi_{ij}^{*}) \neq \pi_1(\chi_{ik}^{*})$. Then either

$$\chi_{ij}^{*} \cap \big[\chi_{ik}^{*} \circ \chi_{kj}^{*}\big] \neq \chi_{ij}^{*} \;\text{ or }\; \chi_{ik}^{*} \cap \big[\chi_{ij}^{*} \circ \chi_{jk}^{*}\big] \neq \chi_{ik}^{*} \,.$$

But this contradicts (4) which is an equality for the fixpoint. Hence these sets coincide for each i and we denote them by $K_i^{*} = \pi_1(\chi_{ij}^{*})$. So far we have shown that if $\widehat{\chi}$ and hence $\widehat{\chi}^{*}$ have any solution $y \in \mathcal{A}(R, K)$ then $y(r_i) \in K_i^{*}$ holds for all i.

In order to prove the existence of solutions we will show that each solution of $\widehat{\chi}^{*}$ on a subgraph of $\mathcal{G}$ can be extended to a solution on the whole graph $\mathcal{G}$: Whenever $\mathcal{G}_1 = (R_1, E_1)$ is an induced subgraph of $\mathcal{G}$ and $y \in \mathcal{A}(R_1, K)$ is a solution of $\widehat{\chi}^{*}$ on $\mathcal{G}_1$, then y can be extended to a solution of $\widehat{\chi}^{*}$ on $\mathcal{G}$. It suffices to prove this for the case that $R_1 = R \setminus r_1$, i.e. $\mathcal{G}_1$ is obtained from $\mathcal{G}$ by deleting the vertex r_1. Let $y \in \mathcal{A}(R_1, K)$ be a solution of $\widehat{\chi}^{*}$ on $\mathcal{G}_1$, i.e.

$$y(r_i) \in K_i^{*},\ \forall r_i \in R_1 \ \text{ and } \ \chi_{ij}^{*}\big[y(r_i), y(r_j)\big] = 1, \ \ \forall r_i, r_j \in R_1 \,. \tag{5}$$

[3] See Remark 1 for a more lucid explanation of the algorithm.

Let $U_i \subset K_1^*$ be the sets

$$U_i = \pi_1 \Big[\big(K_1^* \times y(r_i) \big) \cap \chi_{1i}^* \Big] \ .$$

By Lemma 1 every $\widehat{\chi}^{(t)}$ and therefore $\widehat{\chi}^*$ are closed. Hence the sets U_i are hulls of $\mathcal{U}_{K_1^*}$. Let us assume that y cannot be extended to a solution on $\mathcal{G}$, then

$$\bigcap_{i \neq 1} U_i = \emptyset \ .$$

Hence there must be a pair of vertices i, $j \neq 1$ for which $U_i \cap U_j = \emptyset$ holds. But then

$$\chi_{ij}^* \cap \left[\chi_{i1}^* \circ \chi_{1j}^* \right] \neq \chi_{ij}^* \ ,$$

which contradicts the fixpoint equality. Thus y can be extended to a solution on $\mathcal{G}$. □

Remark 1. In order to give a more lucid interpretation of (4) we start with a closer look on the expression

$$\chi_{ij} \cap \big(\chi_{ik} \circ \chi_{kj} \big) \ .$$

It represents a new predicate on the edge (r_i, r_j) with the following property: A pair of symbols (k_1, k_2) is allowed by this predicate only if

1. It is allowed by χ_{ij}.
2. There is at least one k_3 so that (k_1, k_2, k_3) is a solution on the triangle (r_i, r_j, r_k).

Hence (4) can be interpreted in the following way: For the edge (r_i, r_j) and the current predicate $\chi_{ij}^{(t)}$ we check whether each pair of symbols (k_1, k_2) allowed by $\chi_{ij}^{(t)}$ can be extended to an solution on each triangle (r_i, r_j, r_k) containing the edge (r_i, r_j). If not, this symbol pair is not allowed by $\chi_{ij}^{(t+1)}$. Therefore in the fixpoint each symbol pair allowed on a edge can be extended to a solution on each triangle.

At first glance it may seem that this condition is sufficient for the existence of solutions in the general case (i.e. if K is not necessarily ordered and the predicates are not necessarily closed). Therefore we give here a simple counterexample. Suppose we try to color a fully connected graph with four vertices (tetrahedron) with three colors. It is obviously impossible. But the initial predicates – the colors of two vertices connected by an edge must be different – remain nonempty and stable by applying (4).

Remark 2. The proof of Theorem 1 shows in particular that algorithm (4) gives more than only a "yes" or "no" answer whether $\widehat{\chi}$ has solutions: The sets K_i^* describe the solution set in some weak sense. For each $k \in K_i^*$ there is at least one solution y with $y(r_i) = k$. And more generally, each solution on a subgraph $\mathcal{G}' \subset \mathcal{G}$ fulfilling (5) can be extended to a solution on $\mathcal{G}$.

The same result can be obtained using the following sequential algorithm. This algorithm executes $n-2$ steps ($n = |R|$). At each step a vertex r_i is removed from the actual graph $\mathcal{G}_i$ and the current LCP $\widehat{\chi}^{(i)}$ is replaced by a modified LCP $\widehat{\chi}^{(i+1)}$ on the remaining graph $\mathcal{G}_{i+1}$. The algorithm performs the following operations in order to execute the i-th step:

$$\chi_{jl}^{(i+1)} = \chi_{jl}^{(i)} \cap \left[\chi_{ji}^{(i)} \circ \chi_{il}^{(i)}\right] \quad \forall j, l > i\ . \tag{6}$$

The graph $\mathcal{G}_{n-2}$ obtained after executing $n-2$ steps has two nodes $(r_{n-1},\, r_n)$ and one predicate $\chi_{n-1n}^{(n-2)}$. The LCP $\widehat{\chi}$ has solutions on $\mathcal{G}$ if and only if $\chi_{n-1n}^{(n-2)}$ is nonempty. In this case we can choose any solution of $\chi_{n-1n}^{(n-2)}$ on $\mathcal{G}_{n-2}$ and extend it step by step to a solution of $\widehat{\chi}$ on $\mathcal{G}$. It is easy to prove the existence of such an extension by slightly modifying the proof of Theorem 1. The complexity of the algorithm is less than $n^3|K|^3$.

5 The MaxMin Problem for Closed Similarity Functions

The consistent labeling problem considered so far assumes that symbol pairs on neighbouring vertices have to fulfil some binary constraints. There are many applications where symbol pairs are "ranked" by real and not boolean numbers and local predicates are replaced by real valued functions $f_{ij} \colon K \times K \mapsto \mathbb{R}$ associated with the edges (r_i, r_j) of $\mathcal{G}$. The MaxMin problem is then

$$\max_{y \in \mathcal{A}(R,K)} \min_{(i,j)} f_{ij}\left[y(r_i), y(r_j)\right]\ . \tag{7}$$

We will show how to generalize the results obtained for consistent labeling in order to solve this problem. This is possible if all functions f_{ij} fulfil a closeness property.

Let $B_\epsilon f$ be the *binarization* of a function f with threshold ϵ:

$$\big(B_\epsilon f\big)(k_1, k_2) = \begin{cases} 1 & \text{if } f(k_1,k_2) > \epsilon \\ 0 & \text{otherwise} \end{cases}\ .$$

By binarizing real valued functions f_{ij} we obtain predicates on $K \times K$. Let us assume that K is ordered and binarizations of all f_{ij} are closed predicates for each threshold ϵ. Then the MaxMin problem can be solved as follows:

1. Choose a threshold ϵ and binarize all f_{ij}: $\chi_{ij} = B_\epsilon f_{ij}$.
2. Solve the consistent labeling for $\widehat{\chi}$.
3. If $\widehat{\chi}$ has solutions, then increase ϵ, else decrease ϵ. Go to 1.

This algorithm is not very "nice" and the question arises whether it is possible to "commute" binarization and consistent labeling. This is indeed possible. In order to verify this we use the following operations on functions $f \colon K \times K \mapsto \mathbb{R}$:

$$\begin{aligned} \left[f \oplus g\right](k_1,k_2) &= \min\left[f(k_1,k_2),\, g(k_1,k_2)\right] \\ \left[f \otimes g\right](k_1,k_2) &= \max_k \min\left[f(k_1,k),\, g(k,k_2)\right]\ . \end{aligned}$$

It is easy to prove that:

$$f, g @>\oplus>> f \oplus g$$

$$@V B_\epsilon VV @V B_\epsilon VV$$

$$B_\epsilon f, B_\epsilon g @>\cap>> (B_\epsilon f) \cap (B_\epsilon g) = B_\epsilon(f \oplus G)$$

and

$$f, g @>\otimes>> f \otimes g$$

$$@V B_\epsilon VV @V B_\epsilon VV$$

$$B_\epsilon f, B_\epsilon g @>\circ>> (B_\epsilon f) \circ (B_\epsilon g) = B_\epsilon(f \otimes G)$$

are commutative diagrams. This leads to the following iterative algorithm solving (7)

$$\begin{aligned} f_{ij}^{(0)} &= f_{ij} \\ f_{ij}^{(t+1)} &= f_{ij}^{(t)} \oplus \Big[\bigoplus_k \big(f_{ik}^{(t)} \otimes f_{kj}^{(t)}\big)\Big] \end{aligned} \tag{8}$$

It is clear that this algorithm reaches a fixpoint: $f_{ij}^{(t+1)} = f_{ij}^{(t)}$ $\forall(i,j)$. It follows from (8) that

$$\max_{k_i, k_j} \big[f_{ij}^*(k_i, k_j)\big] = c \quad \forall(i,j)$$

holds for the fixpoint f^*, where c is the optimum of (7). Choosing the the sets

$$K_i^* = \{k_i \mid f_{ij}^*(k_i, k_j) = c\}$$

we obtain the solution of (7) as described in the proof of Theorem 1.

Remark 3. As well as in case of consistent labeling we can solve (7) equivalently by an sequential algorithm. Both algorithms coincide up to substituting (6) by

$$f_{jl}^{(i+1)} = f_{jl}^{(i)} \oplus \big[f_{ji}^{(i)} \otimes f_{il}^{(i)}\big] \, .$$

Again the complexity is less than $n^3|K|^3$.

6 Conclusion

We have shown that consistent labeling can be solved in polynomial time on fully connected graphs if the symbol set is ordered and the local predicates are closed, i.e. preserve a structure derived from the ordering. These results were generalized in order to solve an important optimization problem on labeled graphs – the MaxMin problem. It is also solvable in polynomial time if its similarity functions preserve the aforementioned structure. Whether assumptions of this kind are sufficient in order to solve another prominent representative of optimization problems on labeled graphs – maximization of the sum of similarities (3) – remains an open and very intriguing question.

Applications using the obtained results are subject of forthcoming publications.

References

[1] H.L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput*, 25(6):1305–1317, 1996.

[2] L.S. Davis and A. Rosenfeld. Cooperating processes for low-level vision: A survey. In R. Kasturi and R. Jain, editors, *Computer Vision: Principles*, pages 282–299. IEEE Computer Soc. Press, 1991.

[3] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume 2. Addison–Wesley Publ. Comp., 1993.

[4] R.A. Hummel and S.W. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. PAMI*, 5:267–286, 1983.

[5] A. Rosenfeld, R.A. Hummel, and S.W. Zucker. Scene labeling by relaxation operations. *IEEE Trans. SMC*, 6:420–433, 1976.

[6] M.I. Schlesinger. *Mathematical Methods of Image Processing.* Naukova Dumka, Kiev, 1989.

[7] M.I. Schlesinger and B. Flach. Monotone conjunctive structures of second order. In T. Vintsiuk, editor, *Fourth Ukrainian Conference on Signal/Image Processing and Pattern Recognition*, pages 29–32, 1998.

[8] M.I. Schlesinger and V. Hlaváč. *Deset přednášek z teorie statistického a strukturního rozpoznávání.* Vydavatelství ČVUT, Praha, Czech Republic, 1999.