Czech Pattern Recognition Workshop 2000, Tomáš Svoboda (Ed.)
Peršlák, Czech Republic, February 2–4, 2000
Czech Pattern Recognition Society

Some solvable subclasses of structural recognition problems

M.I. Schlesinger, B. Flach

International Research and Training Centre of Information Technologies and Systems,
National Academy of Science of Ukraine. Email:schles@image.kiev.ua
Institute of Artificial Intelligence, Dresden University of Technology
Email: bflach@ics.inf.tu-dresden.de

Abstract *During the last decades a lot of applied structural recognition problems came to light which might be reduced to consistent labelling problems. The consistent labelling problem and its appropriate fuzzy and probabilistic modifications became a proper formalism for a unified formulation of such problems regardless of their specific applied contents. On the other hand there is no formal construction for a unified solution of these problems because they proved to be not polynomially solvable.*

Despite of the complexity of the whole class of labelling problems there are rather wide subclasses of applied interest which are polynomially solvable. In this paper we describe several examples of such subclasses of labelling problems. All these subclasses have a common property: It is possible to introduce several mechanisms of equivalent transformations of the problems. The algorithm solving a problem has then the form of succesive equivalent transformations of the initial problem until it turns into a problem with a quite evident solution.

1 Survey of the known problems and their generalized formulation

1.1 Consistent labelling problem

Structural recognition is the analysis of complex objects composed of several parts. A comlex object is considered as a finite set T consisting of simple objects t. Every simple object $t \in T$ may stay in some state k from a finite set of states K. A complete description of a complex object T is a function $f\colon T \to K$ assigning each simple object $t \in T$ its state $f(t)$ where the object stays. The function f will be called a *labelling*, whereas elements of the set K are called labels.

Suppose the labelling f satisfies some a priori restrictions, given by a local-conjunctive predicate of second order [5]. That mean a subset $\Omega \subset T \times T$ of pairs object-object is given and a function $g(t_1, t_2)\colon K \times K \to \{0, 1\}$ is assigned to each object pair $(t_1, t_2) \in \Omega$. The value $g(t_1, t_2, k_1, k_2)$ determines whether the object t_1 may stay in the state k_1 when the object t_2 stays in the state k_2. The whole labelling $f\colon T \to K$ is called consistent if

$$\bigwedge_{(t_1,t_2)\in\Omega} g\big(t_1, t_2, f(t_1), f(t_2)\big) = 1 \ . \tag{1}$$

Let us suppose that some observations were made for each object $t \in T$ independently, resulting in additional, a posteriori information about the state of each object. This information narrow the set of possible states for each object and is denoted by functions $q(t)\colon K \to \{0, 1\}$. The value $q(t, k)$ defines, whether the state k is contained in the reduced set of states or not.

The consistent labelling problem [1] consists in answering the question, whether the a posteriori information about the states of the objects is consistent with the a priori information, i.e. the problem consists in calculating the quantity

$$\bigvee_{f\in K^T} \Bigg[\bigg(\bigwedge_{(t_1,t_2)\in\Omega} g\big(t_1, t_2, f(t_1), f(t_2)\big) \bigg) \wedge \wedge \bigg(\bigwedge_{t\in T} q\big(t, f(t)\big) \bigg) \Bigg] \tag{2}$$

and seeking for a labelling f^* for which

$$G(f^*) = \bigg(\bigwedge_{(t_1,t_2)\in\Omega} g\big(t_1, t_2, f^*(t_1), f^*(t_2)\big) \bigg) \wedge \wedge \bigg(\bigwedge_{t\in T} q\big(t, f^*(t)\big) \bigg) = 1 \ . \tag{3}$$

The designation K^T in expression (2) means the set of all possible functions $f\colon T \to K$.

The following notions and assumptions will be necessary for the further considerations. The set Ω is called the structure of the complex object T. Without loss of generality we will assume that the set T is completely ordered so that for every pair of objects $t_1 \neq t_2$ either $t_1 < t_2$ or $t_2 < t_1$ is fulfilled. We will also assume that every pair (t_1, t_2) of the structure Ω satisfies the inequality $t_1 < t_2$.

1.2 Minimax labelling problem

The consistent labelling problem may be slightly generalized so that functions $g(t_1, t_2)$ and $q(t)$ are not binary, but real valued functions. The quality of a labeling function f is defined then as the number

$$G(f) = \min\Big[\min_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big);\; \min_{t\in T} q\big(t,f(t)\big)\Big]\,. \tag{4}$$

The *minimax* labelling problem consists in calculating the quality of the best labelling

$$\max_{f\in K^T}\min\Big[\min_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big);\; \min_{t\in T} q\big(t,f(t)\big)\Big] \tag{5}$$

and in looking for the best labelling

$$f^* = \arg\max_{f\in K^T}\min\Big[\min_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big);\; \min_{t\in T} q\big(t,f(t)\big)\Big]\,. \tag{6}$$

1.3 Maxsum labelling problem

Quite a different labelling problem arises when the quality of labelling is not defined by the expression (4) but by the sum

$$G(f) = \sum_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big) + \sum_{t\in T} q\big(t,f(t)\big)\,. \tag{7}$$

In this case the problem consists in looking for the best labelling

$$f^* = \arg\max_{f\in K^T}\Big[\sum_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big) + \sum_{t\in T} q\big(t,f(t)\big)\Big]\,. \tag{8}$$

A special case of the maxsum problem is well known and widely used: If $T = \{1,2,...,n\}$, and the structure Ω is an one-dimensional chain, i.e. includes only the pairs of the form $(i,i+1)$, $i = 1,2,...,n-1$. The problem (8) can be solved in this case by dynamic programming [6] and is widely used in lots of applied recognition problems.

The problem becomes essentialy more complex when the structure Ω is more complex than a chain. Consequently more powerful mathematical tools than dynamic programming are required in order to solve such problems.

1.4 Analysis of Markov random objects

The functions $g(t_1,t_2)$ and $q(t)$ can be treated as statistical parameters of a Markov random complex object. Then certain statistical problems arising in recognition of such objects are reduced to the calculation of the probability [3, 7]

$$\sum_{f\in K^T}\Big[\Big(\prod_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big)\Big) \times \Big(\prod_{t\in T} q\big(t,f(t)\big)\Big)\Big]\,. \tag{9}$$

1.5 Generalized formulation of the problems

All problems discussed above, can be formulated uniformly by use of the following formal construction.

Let T be a finite set of objects, K be a finite set of labels and $f\colon T\to K$ be a labelling. Let W be a set endowed with two operations: $\oplus\colon W\times W\to W$ and $\otimes\colon W\times W\to W$. The operations have to fulfil the following properties of commutativity, associativity and distributivity

$$\begin{aligned} &a\oplus b = b\oplus a,\\ &a\otimes b = b\otimes a,\\ &a\oplus(b\oplus c) = (a\oplus b)\oplus c,\\ &a\otimes(b\otimes c) = (a\otimes b)\otimes c,\\ &a\otimes(b\oplus c) = (a\otimes b)\oplus(a\otimes c). \end{aligned} \tag{10}$$

It is supposed furthermore that W has identity elements for each of the operations i.e. elements $1^\otimes\in W$ and $0^\oplus\in W$ exist, for which equalities $a\oplus 0^\oplus = a$, $a\otimes 1^\otimes = a$ and $a\otimes 0^\oplus = 0^\oplus$ hold for any $a\in W$. To say it in another way, the set W endowed with the operations $\oplus$ and $\otimes$ forms an algebraic structure of a semiring [2].

Let $\Omega\subset T\times T$ be a second order structure on the set T, $g(t_1,t_2)\colon K\times K\to W$ be functions assigned to pairs $(t_1,t_2)\in\Omega$ and $q(t)\colon K\to W$ be functions assigned to objects $t\in T$. These functions $g(t_1,t_2)$ and $q(t)$ define the quality $G(f)\in W$ of a labelling f as the number

$$\Big[\bigotimes_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big)\Big]\otimes\Big[\bigotimes_{t\in T} q\big(t,f(t)\big)\Big]\,. \tag{11}$$

The problems formulated above consist in calculating the number

$$\bigoplus_{f\in K^T} G(f)\,. \tag{12}$$

The problems (3), (6) and (8) require also to look for a labelling f^* that fulfils

$$G(f^*) = \bigoplus_{f\in K^T} G(f)\,. \tag{13}$$

The consistent labelling problem is a special case of the problem (12) and (13) where operations $(\oplus,\otimes)$ are the logical operations $(\vee,\wedge)$. A minimax labelling problem is formed with operations $(\max,\min)$, maxsum labelling problems (7) and (8) correspond to operations $(\max,+)$, and finally, statistical problems correspond to summation and multiplication in the conventional sense of these words.

In the following sections we describe the solution of some subclasses of the formulated problems. Every subclass under consideration corresponds to either a certain subclass of structures Ω or a certain subclass of functions $g(t_1,t_2)\colon K\times K\to W$.

2 Simple nets. Equivalent transformations of parallel and sequential branches of the structures

If the structure of the problem is a so-called *simple net*, a general algorithm for solving the problem (12) can be formulated for every choice of the operations $\oplus$ and $\otimes$.[1] A simple net is defined in the following recurrent way.

1. A structure Ω consisting of a single pair (t_1, t_2) of objects is a simple net.

2. **Inserting of a new object.** Let Ω form a simple net on T, let t be an object of T and t' be some new object not contained in T. Then the structure $\Omega \cup \{(t, t')\}$ on the set $T \cup \{t'\}$ is also a simple net.

3. **Inserting of a serial branch** Let Ω be a simple net on T, let (t_1, t_2) be a pair contained in Ω and let t' be some new object not contained in T. Then the structure $\big(\Omega \setminus \{(t_1, t_2)\}\big) \cup \{(t_1, t'), (t', t_2)\}$ on $T \cup \{t'\}$ is also a simple net.

4. **Inserting of a parallel branch** Let Ω be a simple net on T, let (t_1, t_2) be a pair contained in Ω and let t' be some new object not contained in T. Then the structure $\Omega \cup \{(t_1, t'), (t', t_2)\}$ on $T \cup \{t'\}$ is also a simple net.

The following algorithm solves the problem (12) if Ω is a simple net. The algorithm transforms the input data step by step. At each step some object t^* is removed from the set T as well as all pairs of objects including t^* are removed from the structure Ω. The functions $q(t)$ for the rest of objects of T are changed as well as the functions $g(t_1, t_2)$ for the rest of pairs of Ω. They are changed so, that the new problem is guaranteed to be equivalent to the original one.

Let T, Ω, $g(t_1, t_2)$ $\forall (t_1, t_2) \in \Omega$ and $q(t)$ $\forall t \in T$ be the current set of objects, structure and functions resp. obtained after the previous step of the algorithm. The current step consits then in any of the following three operations.

1. **Deleting of an object** Let t' be an object which is contained only in a single pair $(t^*, t') \in \Omega$. Then the object t' is removed from T as well as the pair (t^*, t') is removed from Ω. The numbers $q(t^*, k), k \in K$, are changed by the operator

$$q(t^*, k) ::= q(t^*, k) \otimes \left[\bigoplus_{k' \in K} \big(g(t^*, t', k, k') \otimes q(t', k') \big) \right]. \tag{14}$$

The expression (14) is meant just as an operator and not as an equality: the designation $q(t^*, k)$ on the right side of the operator means the value of the corresponding variable before operating and the same designation on the left side means the value of the same variable after operating.

2. **Equivalent transformation of serial branches.** Let t' be an object contained in two and only two pairs (t_1, t'), (t', t_2) and suppose furthermore that the pair (t_1, t_2) is not contained in the structure Ω. Then the object t' is removed from the set T as well as the pairs $(t_1, t'), (t', t_2)$ are removed from Ω, and the pair (t_1, t_2) is included in Ω. The function $g(t_1, t_2)$ for this included pair is defined as follows

$$\begin{aligned} & g(t_1, t_2, k_1, k_2) = \\ & = \bigoplus_{k' \in K} \Big[g(t_1, t', k_1, k') \otimes q(t', k') \otimes g(t', t_2, k', k_2) \Big]. \end{aligned} \tag{15}$$

3. **Equivalent transformation of parallel branches.** Let t' be an object contained in two and only two pairs (t_1, t') (t', t_2) and suppose furthermore that the pair (t_1, t_2) is also contained in the structure. Then the object t' is removed from T, the pairs $(t_1, t'), (t', t_2)$ are removed from Ω, and the numbers $g(t_1, t_2, k_1, k_2)$ are changed according to

$$\begin{aligned} & g(t_1, t_2, k_1, k_2) ::= g(t_1, t_2, k_1, k_2) \otimes \\ & \otimes \left[\bigoplus_{k' \in K} \big(g(t_1, t', k_1, k') \otimes q(t' k') \otimes g(t', t_2, k', k_2) \big) \right]. \end{aligned} \tag{16}$$

It is clear that the algorithm removes just one object out of T at each step. Consequently only one object t^* will remain after $|T| - 1$ steps and the numbers $q(t^*, k^*)$ will have been calculated for this object. The solution of the problem (12) is the number

$$\bigoplus_{k^* \in K} q(t^*, k^*).$$

The total amount of calculations necessary for the solution is of order $|T| \cdot |K|^3$.

The problem (13) can be solved with an algorithm that is very similar to just described one. For this aim some auxiliary data are to be saved on each step (14), (15) or (16) of the algorithm. The collection of these saved data obtained after all steps is the base for constructing the labelling (13).

The class of simple nets is a very small subset of all possible structures. Nevertheless, they essentially increase the tools for structural analysis as compared with the situation when only one-dimensional chains or acyclic structures [4] are considered.

3 Consistent labelling problem for arbitrary structures. Equivalent transformation of stars to simplexes.

The solution of labelling problems for simple nets is very similar to the well-known transformations of parallel and serial connections for electrical cirquits which are widely used in the analysis of electric cirquits . It is well-known however that these transformations are not sufficient for the analysis of arbitrary cirquits. Additional rules of equivalent transformations are necessary to analyse arbitrary cirquits, namely the star-triangle and triangle-star transformations. In this section we will describe similar transformations of structures which will help us to solve labelling problems not only for simple nets but for arbitrary structures. Taking into account the set of all possible structures we will be extorted to restrict the class of problems in some other way.

[1] Of course the operations have to fulfil (10).

Let $T = \{1, 2, ..., n\}$ be the set of objects, K be the finite set of labels and $f: T \to K$ be a labelling. When constructing algorithms for labelling problems in case of simple nets, the structure Ω was restricted very strongly whereas the set of labels K could be arbitrary. Now we will act quite contrary: we will allow any structure and restrict the set K of labels rather strongly. We will assume that the set K is completely ordered: For every pair of labels k_1 and k_2 there is an interval $I(k_1, k_2)$, defined as the set labels k' fulfilling $k_1 \le k' \le k_2$ if $k_1 \le k_2$ or fulfilling $k_1 \ge k' \ge k_2$ if $k_1 \ge k_2$. Furthermore, if $k' \in I(k_1, k_2)$, the label k' will be called the inner label of the triple k', k_1, k_2. It is evident, that every triple of labels has at least one inner label.

Suppose the structure Ω contain all pairs of the type (t_1, t_2), $t_1 \in T$, $t_2 \in T$, $t_1 < t_2$ and the functions $g(t_1, t_2)\colon K \times K \to \{0, 1\}$ fulfil the following properties for each pair $(t_1, t_2) \in \Omega$: Let $k_1, k_1', k_1'', k_2, k_2', k_2''$ be six labels satisfying the condition

$$\begin{aligned} g(t_1, t_2, k_1, k_2) &= g(t_1, t_2, k_1', k_2') = \\ &= g(t_1, t_2, k_1'', k_2'') = 1 \ . \end{aligned}$$

Let furthermore k_1^* be inner label in the triple k_1, k_1', k_1'' and k_2^* be inner label in the triple k_2, k_2', k_2''. Then

$$g(t_1, t_2, k_1^*, k_2^*) = 1 \ .$$

This condition will be called the interval restriction. It should be mentioned that the interval restriction is fulfilled for any function $g(t_1, t_2)\colon K \times K \to \{0, 1\}$ in case of $|K| = 2$.

At last, let us suppose that for every object $t \in T$ a function $q(t)\colon K \to \{0, 1\}$ is defined and this function is not restricted by any limitation.

The following algorithm solves the consistent labelling problem if the interval conditions are fulfilled: The algorithm executes $|T| - 2$ steps. On every step some object is removed from the set T and the numbers $g(t_1, t_2, k_1, k_2)$ are changed for the remaining pairs (t_1, t_2).

Let $\{i, i+1, i+2, ..., n\}$ denote the current objects of T, (t_1, t_2), $i \le t_1 < t_2 \le n$ denote the current pairs of the structure Ω as well as $g(t_1, t_2, k_1, k_2)$ denote the current numbers for the pairs $(t_1, t_2) \in \Omega$ before executing the i-th step. The algorithm performs the following operations in order to execute the i-th step (These operations can be understood as a transformation of a star into a simplex.

1. The object i is removed from the set T.
2. The pairs (i, t), $t > i$ are removed from the set Ω.
3. For all pairs (t_1, t_2), $i < t_1 < t_2 \le n$, new values $g(t_1, t_2, k_1, k_2)$ are calculated:

$$\begin{aligned} &g(t_1, t_2, k_1, k_2) ::= g(t_1, t_2, k_1, k_2) \wedge \\ &\wedge \bigvee_{k_i \in K} \Big(g(i, t_1, k_i, k_1) \wedge q(i, k_i) \wedge g(i, t_2, k_i, k_2) \Big) \ . \end{aligned} \tag{17}$$

After executing $n-2$ steps the array of numbers $g(n-1, n, k_{n-1}, k_n)$ is obtained. The solution of the problem (2) is the number

$$\begin{aligned} \bigvee_{k_{n-1}} \bigvee_{k_n} \Big[&q(n-1, k_{n-1}) \wedge g(n-1, n, k_{n-1}, k_n) \wedge \\ &\wedge q(n, k_n) \Big] \end{aligned} \tag{18}$$

The solution of the problem (3) i.e. the construction of the labelling f^*, is obtained in the following way. If the number (18) is zero then the labelling f^* does not exist. In the opposite case we can choose any two values k_{n-1}^* and k_n^* fulfilling

$$q(n-1, k_{n-1}^*) \wedge g(n-1, n, k_{n-1}^*, k_n^*) \wedge g(n, k_n^*) = 1$$

and set

$$f^*(n-1) = k_{n-1}^* \quad \text{and} \quad f^*(n) = k_n^* \ .$$

Suppose now, that we have already assigned the values $f^*(i+1), f^*(i+2), ..., f^*(n)$ of the labelling. In order to assign a value to the object i we can choose any k_i^* satisfying the condition

$$\left(\bigwedge_{t=i+1}^{n} g(i, t, k_i^*, k_t) \right) \wedge q(i, k_i^*) = 1 \ . \tag{19}$$

Such value dead certain exists. The whole amount of computation is of order $|T|^3 \cdot |K|^3$.

It is not very difficult to modify the described algorithm in order to solve the minimax labelling problem. The interval condition for the minimax problem is modified in the following way.

Let k_1^* be an inner label of the triple k_1, k_1', k_1'' and k_2^* be an inner label of the triple k_2, k_2', k_2''. Then the inequality

$$\begin{aligned} &g(t_1, t_2, k_1^*, k_2^*) \ge \\ &\ge \min \Big[g(t_1, t_2, k_1, k_2), g(t_1, t_2, k_1', k_2'), g(t_1, t_2, k_1'', k_2'') \Big] \end{aligned}$$

must be fulfilled for each pair $(t_1, t_2) \in \Omega$ and every six-tuple $k_1, k_1', k_1'', k_2, k_2', k_2''$ of labels. The algorithm for the minimax labeling problem is the same as the algorithm for the consistent labelling problem apart from the fact that the operation $\vee$ must be replaced by the operation of taking the greater of two numbers as well as the operation $\wedge$ must be replaced by the operation of taking the smaller of two numbers.

4 Maxsum labelling problem

The optimization problems (7) and (8) cannot be solved by sequential removing of variables as described in the previous section because the function

$$\max_k \sum_{t \in T} g\big(k, f(t)\big)$$

cannot be performed in the form

$$\sum_{(t_1,t_2)\in T} g\big(t_1,t_2,f(t_1),f(t_2)\big)\ .$$

Nevertheless, the problems (7) and (8) can be solved but under certain much stronger restrictions than the interval restriction. The algorithm for solving the maxsum problems is also based on equivalent transformations of the given problem into a form with quite evident solutions. In the previous section the transformation of the problem consisted in removing variables. Through such elimination of variables the initial multivariate problem was reduced to an univariate problem that could be solved by an exhaustive testing of all values of the remaining single variable. The solution of maxsum problem is also based on an equivalent transformation of the given problem. This transformation does not reduce the dimensionality of the maxsum problem but nevertheless transforms the initial problem into a trivial one.

4.1 Formulation of the main assumptions

Let T be an ordered set of objects, K be an ordered set of labels and $f\colon T\to K$ be a labelling. Let Ω be the set of all possible pairs of the form (t_1,t_2), $t_1,t_2\in T$, $t_1<t_2$. Suppose that numbers $g(t_1,t_2,k_1,k_2)$ are given for each pair (t_1,t_2) of objects and every pair of labels $k_1,k_2\in K$. As well suppose that numbers $q(t,k)$ are given for each object $t\in T$ and label $k\in K$.

The problem in question consists in looking for the labelling

$$f^* = \arg\max_{f\in K^T}\Bigg[\sum_{(t_1,t_2)\in\Omega} g(t_1,t_2,k_1,k_2) + \\ + \sum_{t\in T} q(t,k)\Bigg]\ . \quad (20)$$

In the following we will show an algorithm that solves this problem if the functions g satisfy the following condition: For each pair of objects $(t_1,t_2)\in\Omega$ and every four labels $k_1\le k_1'$, $k_2\le k_2'$ the following inequality must be fulfilled:

$$g(t_1,t_2,k_1,k_2)+g(t_1,t_2,k_1',k_2')\ge \\ \ge g(t_1,t_2,k_1,k_2')+g(t_1,t_2,k_1',k_2)\ . \quad (21)$$

4.2 Consistent labelling problems and relaxation labelling

For the hereafter analysis of the problem (20) we must return to the problem of consistent labeling, i.e. calculating the number

$$\bigvee_{f\in K^T}\Bigg[\bigg(\bigwedge_{(t_1,t_2)\in\Omega}\widetilde{g}\big(t_1,t_2,f(t_1),f(t_2)\big)\bigg)\wedge \\ \wedge\bigg(\bigwedge_{t\in T}\widetilde{q}(t,k)\bigg)\Bigg] \quad (22)$$

and looking for a labelling f^* that satisfies the condition

$$\bigg(\bigwedge_{(t_1,t_2)\in\Omega}\widetilde{g}\big(t_1,t_2,f(t_1),f(t_2)\big)\bigg)\wedge\bigg(\bigwedge_{t\in T}\widetilde{q}(t,k)\bigg)\ , \quad (23)$$

where $\widetilde{g}(t_1,t_2,k_1,k_2)$ and $\widetilde{q}(t,k)$ are binary numbers. These numbers are assumed to satisfy the following condition: for each pair $(t_1,t_2)\in\Omega$ and every four labels $k_1<k_1'$, $k_2<k_2'$ the following inequality holds:

$$\widetilde{g}(t_1,t_2,k_1,k_2)\wedge\widetilde{g}(t_1,t_2,k_1',k_2')\ge \\ \ge\widetilde{g}(t_1,t_2,k_1,k_2')\wedge\widetilde{g}(t_1,t_2,k_1',k_2)\ . \quad (24)$$

This condition will be called monotonous interval condition. It is stronger than the interval condition that enables the solution of the problems (22) and (23) in the previous section. Consequently, the problems may be solved by successive star-simplex transformations. But under monotonous interval condition (24) the problem can be solved also with the simpler algorithm of relaxation labelling [1].

Relaxation labelling consists in the following repeated operations which decrease the numbers $\widetilde{g}(t_1,t_2,k_1,k_2)$ and $\widetilde{q}(t,k)$:

1. **Exclusion of a pair** (t,k)**.** Let the pair (t,k) be such that

$$\big(\exists t'>t\colon \widetilde{g}(t,t',k,k')=0\ \forall k'\in K\big)\vee \\ \vee\big(\exists t'<t\colon \widetilde{g}(t',t,k',k)=0\ \forall k'\in K\big)\ .$$

Then the variable $\widetilde{q}(t,k)$ is set to 0.

2. **Exclusion of a four-tuple** (t,t',k,k')**.** Let the pair (t,k) be such that $q(t,k)=0$. Then for every $t'<t$ and for every $k'\in K$ the variable $\widetilde{g}(t',t,k',k)$ is set to 0. The variables $\widetilde{g}(t,t',k,k')$, $t'>t$, $k'\in K$, are also set to 0.

Relaxation labelling stops if neither variables $\widetilde{q}(t,k)$ nor $\widetilde{g}(t_1,t_2,k_1,k_2)$ decrease. Let us denote by $\widetilde{g}^*(t_1,t_2,k_1,k_2)$ and $\widetilde{q}^*(t,k)$ the values of the corresponding variables after the relaxation labelling has stopped. The result of relaxation labelling will be called a zero result if $q(t,k)=0$ for every $t\in T$ and $k\in K$ and a non-zero result if $\bigvee_{k\in K} q(t,k)=1$ for every $t\in T$. It is known and quite evident that the non-zero result of relaxation labelling is a necessary condition for the existence of consistent labellings f^* satisfying the condition (23).

It is very important for the solution of the optimization problem (20), that under condition (24) a non-zero result of the relaxation labelling is not only necessary but also sufficient for the existence of consistent labellings f^*. For example, the following labelling is consistent:

$$f^*(t)=\max_{k\in K(t)} k, \quad (25)$$

where $K(t)=\{k\in K \bigm| \widetilde{q}^*(t,k)=1\}$.

4.3 Trivial maxsum problems

Trivial maxsum problems of the type

$$f^* = \arg\max_{f\in K^T}\Bigg[\sum_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big) + \\ + \sum_{t\in T} q(t,k)\Bigg] \quad (26)$$

are defined via the following (in general, wrong!) algorithm for their solution:

1. Determine all label pairs k_1, k_2 which maximize the functions $g(t_1,t_2,k_1,k_2)$ and denote the result for each pair of objects $(t_1,t_2) \in \Omega$ by the binary valued functions

$$\widetilde{g}(t_1,t_2,k_1,k_2) = \begin{cases} 1 & \text{if} \quad g(t_1,t_2,k_1,k_2) = g(t_1,t_2) \\ 0 & \text{else.} \end{cases}$$

where $g(t_1,t_2) = \max_{k_1,k_2} g(t_1,t_2,k_1,k_2)$.
2. Determine all labels k which maximize the functions $q(t,k)$ and denote the result for each object $t \in T$ by the binary valued functions

$$\widetilde{q}(t,k) = \begin{cases} 1 & \text{if} \quad q(t,k) = q(t) \\ 0 & \text{else.} \end{cases}$$

where $q(t) = \max_k q(t,k)$.
3. Solve the consistent labelling problem for the functions $\widetilde{g}$, $\widetilde{q}$ i.e. search for a labelling f^* fulfilling

$$\left(\bigwedge_{(t_1,t_2)\in\Omega} \widetilde{g}\big(t_1,t_2,f^*(t_1),f^*(t_2)\big) \right) \wedge \\ \wedge \left(\bigwedge_{(t\in T} \widetilde{q}\big(t,f^*(t)\big) \right) = 1 \; .$$

by computing the functions $\widetilde{g}^*(t_1,t_2,k_1,k_2)$ and $\widetilde{q}^*(t,k)$.
4. If a zero-result is obtained, the initial maxsum problem is defined as a non-trivial one and its solution is not found.
5. If a non-zero result is obtained, the problem is trivial by definition. In this case there exists at least one consistent labeling and this labelling is also a solution of the maxsum problem. One of the consistent labellings can be obtained for example by the above-mentioned method (25).

The main idea of the solution of the problem (26) is that under monotonous interval condition any given maxsum problem can be equivalently transformed into a trivial maxsum problem.

4.4 Equivalent maxsum problems

Let $g(t_1,t_2,k_1,k_2)$ and $q(t,k)$ be the initial numbers which define the quality of a labeling f

$$\sum_{(t_1,t_2)\in\Omega} g\big(t_1,t_2,f(t_1),f(t_2)\big) + \sum_{t\in T} q(t,k). \tag{27}$$

Let $\varphi(k,t,t')$, $k \in K$, $t' \in T\backslash\{t\}$ be some array Φ of numbers defining new functions g' and q':

$$g'(t_1,t_2,k_1,k_2) = \\ = g(t_1,t_2,k_1,k_2) + \varphi(k_1,t_1,t_2) + \varphi(k_2,t_2,t_1) \\ q'(t,k) = q(t,k) - \sum_{t'\in T\backslash\{t\}} \varphi(k,t,t') \; .$$

These new numbers also define the quality of a labelling f as

$$\sum_{(t_1,t_2)\in\Omega} g'\big(t_1,t_2,f(t_1),f(t_2)\big) + \sum_{t\in T} q'\big(t,f(t)\big) \; . \tag{28}$$

For each labelling f the qualities (27) and (28) are exactly the same. Therefore the optimization problem (26) with numbers $g(t_1,t_2,k_1,k_2)$ and $q(t,k)$ is equivalent to the problem of the same form but with the numbers $g'(t_1,t_2,k_1,k_2)$ and $q'(t,k)$. The class of problems which are equivalent to some initial problem is formed with various arrays Φ containing the numbers $\varphi(k,t,t')$, $k \in K, t \in T, t' \in T\backslash\{t\}$. The main result that enables the solution of maxsum problem under the condition (21) is that every class of equivalent maxsum problems contains at least one trivial problem.

4.5 Solution of maxsum problem

A solution of the maxsum problem is based on the following result.

If the condition (21) is satisfied, every maxsum problem can be transformed to a trivial problem by such numbers $\varphi(k,t,t')$ which minimize the value

$$\sum_{(t_1,t_2)\in\Omega} \max_{k_1,k_2\in K} \Big[g(t_1,t_2,k_1,k_2) + \\ + \varphi(k_1,t_1,t_2) + \varphi(k_2,t_2,t_1) \Big] + \\ + \sum_{t\in T} \max_{k\in K} \Big[q(t,k) - \sum_{t'\in T\backslash\{t\}} \varphi(k,t,t') \Big] \; . \tag{29}$$

The problem (29) is in turn a linear optimization problem. Really, it consists in looking for such numbers $H(t_1,t_2)$, $(t_1,t_2) \in \Omega$, $h(t)$, $t \in T$, $\varphi(k,t,t')$, $k \in K$, $t \in T$ $t' \in T\backslash\{t\}$ which minimize the linear function

$$\sum_{(t_1,t_2)\in\Omega} H(t_1,t_2) + \sum_{t\in T} h(t) \tag{30}$$

under the linear restrictions

$$H(t_1,t_2) - \varphi(k_1,t_1,t_2) - \varphi(k_2,t_2,t_1) \geq \\ \geq g(t_1,t_2,k_1,k_2),\, (t_1,t_2) \in \Omega,\, k_1 \in K,\, k_2 \in K; \tag{31}$$

$$h(t) + \sum_{t\in T\backslash\{t\}} \varphi(k,t,t') \geq q(k,t) \; . \tag{32}$$

Therefore (30) can be solved using the well known methods of linear optimization [8]. Due to the peculiarity of the problem it is however preferable to solve it with a specific and more appropriate algorithm which will be faster than general optimization algorithms.

References

[1] A.Rosenfeld, R.Hummel, and S.Zucker. Scene labeling by relaxation operation. *IEEE Trans. Syst. Man, and Cybern*, 6:420–433, 1976.

[2] E.Fried. *Absztrakt algebra - elemi ueton*. Mueszaki Koenyvkiadoe, Budapest, 1972.

[3] M.B.Averintsev. Description of markov random fields by means of gibbs conditional probabilities. *Theory of Probabilities and its Applications*, 17(1):21–35, 1972. in russian.

[4] M.I.Schlesinger and V.Hlavac. *Deset prednasek z teorie statistickeho a strukturniho rozpoznavani*. Vydavatelstvi CVUT, Praha, 1999.

[5] M.Minsky and S.Papert. *Perceptrons: An introduction to Computational Geometry*. MIT Press, Cambridge, Mass., 2 edition, 1988.

[6] R.E.Bellman and S.E.Dreyfus. *Applied Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1962.

[7] S.Geman and D.Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell*, 6:721–741, 1984.

[8] S.I.Zuchovickij and L.I.Avdejeva. *Linejnoje i vypukloje programmirovanije*. Nauka, Moskva, 1967.