DOKTORANDSKÉ DNY 2006

sborník workshopu doktorandů FJFI
oboru Matematické inženýrství

10. a 24. listopadu 2006

P. Ambrož, Z. Masáková (editoři)

Seznam příspěvků

Předmluva

Katedry matematiky a fyziky zajišťují studijní obor Matematické inženýrství doktorského studijního programu Aplikace přírodních věd, který je akreditován na Fakultě jaderné a fyzikálně inženýrské ČVUT. Školitelé z katedry matematiky, katedry fyziky a z různých ústavů AV ČR vychovávají doktorandy v zaměřeních matematické modelování, matematická fyzika a softwarové inženýrství. Disertační práce studentů těchto zaměření pokrývají široký okruh témat, od matematických modelů přírodních procesů, přes problémy kvantové teorie, až po neuronové sítě, či databázové systémy.

Doktorandské dny, které se budou počínaje akademickým rokem 2006-07 pravidelně konat vždy v měsíci říjnu nebo listopadu, jsou příležitostí pro studenty doktorského studia představit formou prezentace na workshopu svou práci za uplynulý rok nejen svým kolegům doktorandům, oborové radě, ale i všem zájemcům z řad odborné veřejnosti. Uchování příspěvků ve sborníku pak umožní sledovat postup práce jednotlivých doktorandů na jejich vědeckém úkolu.

Věříme, že konání workshopu přispěje ke kvalitě výchovy doktorandů oboru Matematické inženýrství.

Editoři

All about Infinite Words Associated with Quadratic Non-simple Parry Numbers

Ľubomíra Balková

2nd year of PGS, email: `l.balkova@centrum.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Zuzana Masáková, Katedra matematiky, Fakulta jaderná a fyzikálně
inženýrská, ČVUT

Abstract. Studying of factor complexity, palindromic complexity, and return words of infinite aperiodic words is an interesting combinatorial problem. Moreover, investigation of infinite words associated with β-integers $\mathbb{Z}_\beta$, for β being a Pisot number, can be interpreted as investigation of one-dimensional quasicrystals.

In this paper, new results concerning the above combinatorial characteristics for quadratic non-simple Parry number β will be presented. This is the only case among (non-simple) Parry numbers worth of studying palindromes since for non-quadratic cases, there is only a finite number of palindromes in the associated infinite word u_β. We have investigated factor and palindromic complexity, return words, and arithmetics using methods which can be applied for any infinite aperiodic words being fixed points of a substitution.

Abstrakt. Studium komplexity, palindromické komplexity a "return" slov v nekonečných aperiodických slovech je zajímavý kombinatorický problém. Zkoumání nekonečných slov přidružených β-celým číslům lze navíc interpretovat jako zkoumání jednodimenzionálních kvazikrystalů.

V tomto článku představíme nové výsledky týkající se výše zmiňovaných charakteristik a také aritmetiky. Námi zkoumaný případ je jediným případem mezi parryovskými čísly, kdy přidružené nekonečné slovo obsahuje nekonečně mnoho palindromů a je tedy zajímavé z hlediska vyšetřování palindromické komplexity. Použité metody se dají aplikovat na celou řadu slov, která jsou pevnými body substitucí.

1 Introduction

Some kinds of infinite aperiodic words can serve as models for one dimensional quasicrystals, i.e., materials with long-range orientational order and sharp diffraction images of non-crystallographic symmetry. We will focus on infinite words u_β associated with β-integers $\mathbb{Z}_\beta \subset \mathbb{R}$. It has been shown that for β being a Pisot number ($\beta > 1$ being an algebraic integer such that all its Galois conjugates have modulus strictly less than one), $\mathbb{Z}_\beta$ is a uniformly discrete and relatively dense set (in one word, it is a Delone set [10]) fulfilling $\mathbb{Z}_\beta - \mathbb{Z}_\beta \subset \mathbb{Z}_\beta + F$ for a finite set F (the Meyer property [11]). Since self-similar Delone sets fulfilling the Meyer property are suitable models for quasicrystalline structure, β-integers for β being a Pisot number serve as models for one dimensional quasicrystals.

2   Preliminaries

First, let us introduce our "language" which will be used throughout this paper. An *alphabet* $\mathcal{A}$ is a finite set of symbols called *letters*. A concatenation of letters is a *word*. The set $\mathcal{A}^*$ of all finite words (including the empty word ε) provided with the operation of concatenation is a free monoid. We will deal also with right-sided infinite words $u = u_0 u_1 u_2 \cdots$. A finite word w is called a *factor* of the word u (finite or infinite) if there exist a finite word $w^{(1)}$ and a word $w^{(2)}$ (finite or infinite) such that $u = w^{(1)} w w^{(2)}$. The word w is a *prefix* of u if $w^{(1)} = \varepsilon$ and it is a *suffix* of u if $w^{(2)} = \varepsilon$. A concatenation of k letters a (or words a) will be denoted by a^k, a concatenation of infinitely many letters a (or words a) by a^ω. An infinite word u is said to be *eventually periodic* if there exist words v, w such that $u = vw^\omega$. A word which is not eventually periodic is called *aperiodic*. An infinite word u is *uniformly recurrent* if for any $n \in \mathbb{N}$ there exists an $R(n) \in \mathbb{N}$ such that any factor of u of length $R(n)$ contains all factors of length n. The *language* on u is the set of all factors of a word u. A mapping φ on the free monoid $\mathcal{A}^*$ is called a morphism if $\varphi(vw) = \varphi(v)\varphi(w)$ for all $v, w \in \mathcal{A}^*$. Obviously, for determining any morphism it suffices to give $\varphi(a)$ for all $a \in \mathcal{A}$. The action of a morphism can be naturally extended on right-sided infinite words by the prescription

$$\varphi(u_0 u_1 u_2 \cdots) := \varphi(u_0)\varphi(u_1)\varphi(u_2)\cdots .$$

A non-erasing morphism φ, for which there exists a letter $a \in \mathcal{A}$ such that $\varphi(a) = aw$ for some non-empty word $w \in \mathcal{A}^*$, is called a substitution. An infinite word u such that $\varphi(u) = u$ is called a fixed point of the substitution φ. Obviously, every substitution has at least one fixed point, namely

$$\lim_{n \to \infty} \varphi^n(a) .$$

A substitution φ is primitive if there exists an integer exponent k such that for each pair of letters $a, b \in \mathcal{A}$, the letter a appears in the word $\varphi^k(b)$. Queffélec [13] showed that any fixed point of a primitive substitution is a uniformly recurrent infinite word.

3   Beta-expansions and beta-integers

Let $\beta > 1$ be a real number and let x be a positive real number. Any convergent series of the form:

$$x = \sum_{i=-\infty}^{k} x_i \beta^i ,$$

where $x_i \in \mathbb{N}$, is called a *β-representation* of x. As well as it is usual for the decimal system, we will denote the β-representation of x by

$$x_k x_{k-1} \cdots x_0 \bullet x_{-1} \cdots ,$$

if $k \geq 0$, otherwise

$$0 \bullet \underbrace{00 \cdots 00}_{(-1-k) \text{ times}} x_k x_{k-1} \cdots .$$

If a β-representation ends with infinitely many zeros, it is said to be finite and the ending zeros are omitted. A representation of x can be obtained by the following greedy algorithm: There exists $k \in \mathbb{Z}$ such that $\beta^k \leq x < \beta^{k+1}$. Let $x_k := \lfloor \frac{x}{\beta^k} \rfloor$ and $r_k := \{ \frac{x}{\beta^k} \}$, where $\lfloor . \rfloor$ denotes the lower integer part and $\{.\}$ denotes the fractional part. For $i < k$, put $x_i := \lfloor \beta r_{i+1} \rfloor$ and $r_i := \{ \beta r_{i+1} \}$. The representation obtained by the greedy algorithm is called β-*expansion* of x and denoted $\langle x \rangle_\beta$. If $x = \sum_{i=-\infty}^{k} x_i \beta^i$ is the β-expansion of a nonnegative number x, then $\sum_{i=-\infty}^{-1} x_i \beta^i$ is called the β-fractional (or simply fractional) part of x. Let us introduce some important notions connected with β-expansions:

- The set of nonnegative numbers with vanishing fractional part are called nonnegative β-integers, formally

$$\mathbb{Z}_\beta^+ := \left\{ x \geq 0 \;\middle|\; \langle x \rangle_\beta = x_k x_{k-1} \cdots x_0 \bullet \right\}.$$

- The set of β-integers is then defined by

$$\mathbb{Z}_\beta := -\mathbb{Z}_\beta^+ \cup \mathbb{Z}_\beta^+.$$

- All the real numbers with a finite β-expansion of $|x|$ form the set $Fin(\beta)$, formally

$$Fin(\beta) := \bigcup_{n \in \mathbb{N}} \frac{1}{\beta^n} \mathbb{Z}_\beta.$$

- For any $x \in Fin(\beta)$, we denote by $fp_\beta(x)$ the length of its fractional part, i.e.,

$$fp_\beta(x) = \min\{ l \in \mathbb{N} \;\middle|\; \beta^l x \in \mathbb{Z}_\beta \}.$$

The sets $\mathbb{Z}_\beta$ and $Fin(\beta)$ are generally not closed under addition and multiplication. The following notion is important for studying of lengths of the fractional parts which may appear as a result of addition and multiplication.

- $L_\oplus(\beta) := \min\{ L \in \mathbb{N} \mid x, y \in \mathbb{Z}_\beta, x + y \in Fin(\beta) \Longrightarrow fp_\beta(x+y) \leq L \}$.

- $L_\otimes(\beta) := \min\{ L \in \mathbb{N} \mid x, y \in \mathbb{Z}_\beta, xy \in Fin(\beta) \Longrightarrow fp_\beta(xy) \leq L \}$.

If such $L \in \mathbb{N}$ does not exist, we set $L_\oplus(\beta) := \infty$ or $L_\otimes(\beta) := \infty$.

The Rényi expansion of unity simplifies the description of elements of $\mathbb{Z}_\beta$ and $Fin(\beta)$. For its definition, we introduce the transformation $T_\beta(x) := \{\beta x\}$ for $x \in [0, 1]$. The *Rényi expansion of unity* in the base β is defined as

$$d_\beta(1) = t_1 t_2 t_3 \cdots, \quad \text{where} \quad t_i := \lfloor \beta T_\beta^{i-1}(1) \rfloor.$$

One can show that every real number β can be characterized by its Rényi expansion of unity (see [12]). Numbers with a finite Rényi expansion of unity are called *simple Parry numbers*. Numbers with an eventually periodic Rényi expansion of unity are called *(non-simple) Parry numbers*.

4   Infinite words associated with beta-integers

In [15], it is shown that the distances occurring between neighbors of $\mathbb{Z}_\beta$ form the set $\{\Delta_k \mid k \in \mathbb{N}\}$, where

$$\Delta_k := \sum_{i=1}^{\infty} \frac{t_{i+k}}{\beta^i} \text{ for } k \in \mathbb{N}. \tag{1}$$

It is evident that the set $\{\Delta_k \mid k \in \mathbb{N}\}$ is finite if and only if $d_\beta(1)$ is eventually periodic.

Every Pisot number, i.e., a real algebraic integer greater than 1, all of whose conjugates are of modulus strictly less than 1, is a Parry number. On the other hand, every quadratic Parry number is Pisot. This explains that if we deal with quadratic Parry numbers β, the set $\mathbb{Z}_\beta$ models one-dimensional quasicrystals.

From now on, we will restrict our considerations to quadratic Parry numbers. The Rényi expansion of unity for a simple quadratic Parry number β is equal to $d_\beta(1) = pq$, where $p \geq q$, in other words, β is the positive root of the polynomial $x^2 - px - q$. Whereas the Rényi expansion of unity for a non-simple quadratic Parry number β is equal to $d_\beta(1) = pq^\omega$, where $p > q \geq 1$, and β is the greater root of the polynomial $x^2 - (p+1)x + p - q$. Drawn on the real line, there are only two distances between neighboring points of $\mathbb{Z}_\beta$. The longer distance is always $\Delta_0 = 1$, the smaller one is Δ_1.

If we assign letters 0, 1 to the two types of distances Δ_0 and Δ_1, respectively, and write down the order of distances in $\mathbb{Z}_\beta^+$ on the real line, we naturally obtain an infinite word; we will denote this word by u_β. Since $\beta\mathbb{Z}_\beta^+ \subset \mathbb{Z}_\beta^+$, it can be shown easily that the word u_β is a fixed point of a certain substitution φ (see e.g. [7]); in particular, for the non-simple quadratic Pisot number β, the generating substitution is

$$\varphi(0) = 0^p 1, \quad \varphi(1) = 0^q 1. \tag{2}$$

5   Combinatorial and arithmetical properties of infinite words

To understand the physical properties of quasicrystals, it is useful to investigate the combinatorial and arithmetical properties of the infinite aperiodic words u_β modeling quasicrystals.

- The number of local configurations of atoms in quasicrystalline materials is described by *factor complexity*. It is a function associating to every integer n the number of different factors of length n contained in u_β.

- The local symmetry of the material corresponds to *palindromic complexity*. It is a function associating to every integer n the number of different palindromes of length n contained in u_β, where *palindrome* is a word which stays the same when read backwards.

- Another interesting characterization of richness of motives appearing in quasicrystals is given by the notion of *return words*. Let w be a factor of u_β. Take an arbitrary occurrence of w in u_β. You obtain a return word of w if you read letters successively, beginning at the first letter of w and ending with the letter preceding the very next occurrence of w.

There is one more important reason why to deal with β-integers. It is sometimes useful in computer science to consider addition or multiplication in β-arithmetics. $\mathbb{Z}_\beta$ is generally not closed under addition and multiplication, thus it is useful to study the fractional parts that may appear as results of these operations and to estimate their lengths, i.e., to find the values of $L_\oplus(\beta)$ and $L_\otimes(\beta)$.

6 Summary of known results

Let us remind that all the characteristics we consider here, i.e., factor complexity, palindromic complexity, return words, and arithmetics, have been already investigated for quadratic Parry units.

- Factor complexity of the infinite word u_β associated with β being a quadratic Parry unit is equal to $n + 1$ [5], i.e., u_β in this case is a Sturmian word. Let us mention that all the eventually periodic words have complexity less or equal to n for some n, hence Sturmian words are the simplest aperiodic words.

- It has been shown in [6] that there are exactly 2 palindromes of any odd length and one palindrome of any even length, hence palindromic complexity P reaches the following values

$$P(2n) = P(0) = 1 \quad \text{and} \quad P(2n + 1) = P(1) = 2 \quad \text{for all } n \in \mathbb{N}. \tag{3}$$

- According to [16] it holds that a uniformly recurrent infinite word on a binary alphabet is Sturmian if and only if for any factor w, there exist two return words of w.

- Results for arithmetics of quadratic Parry units have been found in [5]. For the case of β having the Rényi expansion of unity $d_\beta(1) = p1$, the exact values of $L_\oplus(\beta)$ and $L_\otimes(\beta)$ are $L_\oplus(\beta) = L_\otimes(\beta) = 2$, while for β having the Rényi expansion of unity $d_\beta(1) = p(p-1)^\omega$, it holds $L_\oplus(\beta) = L_\otimes(\beta) = 1$.

A lot of work has been done also for simple Parry numbers. The exact formula for factor complexity of u_β for β being a simple Parry number with the Rényi expansion of unity $d_\beta(1) = t_1 t_2 \cdots t_m$, where $t_1 = t_2 = \cdots = t_{m-1}$ or $t_1 > \max\{t_2, \ldots, t_{m-1}\}$ has been derived in [8]. It is useful to consider palindromic complexity only for the case of $t_1 = t_2 = \cdots = t_{m-1}$. Otherwise, the language of u_β is not closed under reversal and, hence, contains only a finite number of palindromes. The exact formula for palindromic complexity of u_β associated with a simple Parry number with the Rényi expansion of unity $d_\beta(1) = t_1 t_2 \cdots t_m$, where $t_1 = t_2 = \cdots = t_{m-1}$, has been found in [1].

In [9], one can find very precise estimates on $L_\oplus(\beta)$ for β being a quadratic simple Parry number and also some rough estimates on $L_\oplus(\beta)$ for β being a non-simple quadratic Parry number.

7 New results

We will present our results concerning all the characteristics of the infinite word u_β associated with beta-integers for β being a quadratic non-simple non-unit Parry number, i.e., β having the Rényi expansion of unity $d_\beta(1) = pq^\omega$, where $p - 1 > q \geq 1$, in other words, β being the larger root of the polynomial $x^2 - (p+1)x + (p-q)$. In this case, u_β is the fixed point of the substitution $\varphi(0) = 0^p1$, $\varphi(1) = 0^q1$. We have investigated factor [2] and palindromic complexity [3], return words, and arithmetics [4]. We have found the exact values of factor complexity C, which implies that $C(n+1) - C(n) \in \{1, 2\}$ for all $n \in \mathbb{N}$. We have derived the exact values of palindromic complexity P, which confirms that $P(n) \in \{1, 2, 3, 4\}$ for all $n \in \mathbb{N}$. We have shown that for any factor w of u_β, there exist either 2 or 3 return words of w. In the arithmetics, the upper bound on the number $L_\oplus(\beta)$ reached in [9] has been improved. We have shown that $\left\lfloor \frac{p-1}{q} \right\rfloor \leq L_\oplus(\beta) \leq \left\lceil \frac{p}{q} \right\rceil$. Let us describe in more details factor complexity of u_β and the cardinality of the set of return words of u_β.

7.1 Factor complexity

For proofs and precise statements of this section see [2]. To describe complexity of infinite uniformly recurrent words, one can limit his considerations to description of left special factors. In our case, u_β is a fixed point of a primitive substitution, consequently, u_β is uniformly recurrent. Let us remind that a factor w of u_β is left special if both $0w$ and $1w$ are factors of u_β.

Observation 1. *Let us denote by M_n the set of all left special factors of length n of an infinite uniformly recurrent word over a two-letter alphabet. Then the first difference of complexity satisfies*
$$\Delta C(n) = C(n+1) - C(n) = \#M_n.$$

To describe all the left special factors of u_β, let us distinguish more types of them.

Definition 2. Let u_β be the infinite word associated with $d_\beta(1) = pq^\omega$, $p - 1 > q \geq 1$.

- A left special factor $w \in \mathcal{L}(u_\beta)$ is called maximal if neither $w0$ nor $w1$ are left special.

- An infinite word v is called an infinite left special factor of u_β if each prefix of v is a left special factor of u_β.

- A factor w of u_β is called total bispecial if both $w0$ and $w1$ are left special factors of u_β.

Example 3. Let us illustrate a few of left special factors of

$$u_\beta = 0001000100010100010\ldots$$

being the fixed point of the substitution $\varphi(0) = 0001, \varphi(1) = 01$ by construction of the head of a tree containing left special factors. Beginning from the empty word to the

right, one can read all left special factors of length $n \in \{1, 2, \ldots, 14\}$. There are two maximal left special factors 00, 01000100010 and two total bispecial factors 0, 0100010 having length < 14.

Obviously, every left special factor is a prefix of a maximal or an infinite left special factor. Let us describe all the maximal and infinite left special factors:

- All maximal left special factors have the form:

$$U^{(1)} = 0^{p-1}, \tag{4}$$

$$U^{(n)} = T(U^{(n-1)}) = 0^q 1 \varphi(U^{(n-1)}) 0^q \quad \text{for } n \geq 2.$$

- All total bispecial factors have the form:

$$V^{(1)} = 0^q, \tag{5}$$

$$V^{(n)} = T(V^{(n-1)}) = 0^q 1 \varphi(V^{(n-1)}) 0^q.$$

Moreover, $V^{(n-1)}$ is a prefix of $V^{(n)}$ and $V^{(n)}$ is a prefix of $U^{(n)}$ for all $n \in \mathbb{N}$.

- There exists one infinite left special factor of the form $\lim_{n \to \infty} V^{(n)}$.

For n such that

$$|V^{(k)}| < n \leq |U^{(k)}| \quad \text{for some } k \in \mathbb{N},$$

there exist two left special factors of length n. The lengths $|V^{(k)}|, |U^{(k)}|$ play an essential role for determining of complexity. One can easily obtain the recursive formulae of $|V^{(k)}|, |U^{(k)}|$ using their definition (4) and (5).

Combining all the obtained results, we can determine complexity.

Theorem 4. *Let u_β be the fixed point of the substitution $\varphi(0) = 0^p 1$, $\varphi(1) = 0^q 1$, $p - 1 > q \geq 1$. Then for all $n \in \mathbb{N}$*

$$\triangle C(n) = C(n+1) - C(n) = \begin{cases} 2 & |V^{(k)}| < n \leq |U^{(k)}| \quad \text{for some } k \in \mathbb{N}, \\ 1 & \text{otherwise.} \end{cases}$$

7.2 Return words

Let us start with the exact definition of a return word. Let w be a factor of an infinite word $u = u_0 u_1 \cdots$ (with $u_j \in \mathcal{A}$), the length $|w| = \ell$. An integer j is an *occurrence* of w in u if $u_j u_{j+1} \cdots u_{j+\ell-1} = w$. Let j, k, $j < k$, be successive occurrences of w. Then $u_j u_{j+1} \cdots u_{k-1}$ is a *return word* of w. The set of all return words of w is denoted by $M(w)$, i.e.,

$$M(w) = \{u_j u_{j+1} \ldots u_{k-1} \mid j, k \text{ being successive occurrences of } w\}.$$

It is not difficult to see that the set of return words of w is finite for any factor w if u is a uniformly recurrent word.
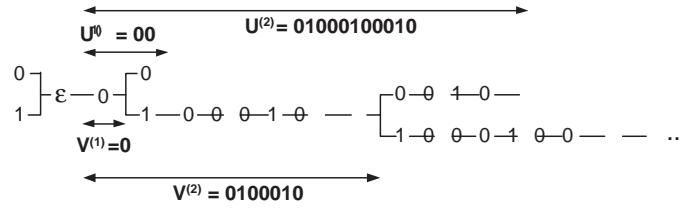
Figure 1: Illustration of the tree of left special factors for u_β being the fixed point of the substitution $\varphi(0) = 0001, \varphi(1) = 01$. We can see total bispecial factors $V^{(k)}$ and maximal left special factors $U^{(k)}$ for $k = 1, 2$.

Example 5. Let $u_\beta = 001001010010010100101 \cdots$ be the fixed point of the substitution $\varphi(0) = 001, \varphi(1) = 01$. Let us show examples of return words:

$$M(0) = \{0, 01\},$$

$$M(00) = \{001, 00101\},$$

$$M(001) = \{001, 00101\},$$

$$M(0010) = \{001, 00101\}.$$

In order to study return words $M(w)$ of factors w of an infinite uniformly recurrent word u, it is possible to limit our considerations to bispecial factors. Namely, if a factor w is not right special, i.e., if it has a unique right extension $a \in \mathcal{A}$, then the sets of occurrences of w and wa coincide, and

$$M(w) = M(wa).$$

If a factor w has a unique left extension $b \in \mathcal{A}$, then $j \geq 1$ is an occurrence of w in the infinite word u if and only if $j - 1$ is an occurrence of bw. This statement does not hold for $j = 0$. Nevertheless, if u is a uniformly recurrent infinite word, then the set $M(w)$ of return words of w stays the same no matter whether we include the return word corresponding to the prefix w of u or not. Consequently, we have

$$M(bw) = bM(w)b^{-1} = \{bvb^{-1} \mid v \in M(w)\},$$

where bvb^{-1} means that the word v is prolonged to the left by the letter b and it is shortened from the right by erasing the letter b (which is always the suffix of v for $v \in M(w)$).

For an aperiodic uniformly recurrent infinite word u, each factor w can be extended to the left and to the right to a bispecial factor. To describe the cardinality of $M(w)$, it suffices therefore to consider bispecial factors w.

Observation 6. *Let w be a bispecial factor of u_β containing at least one 1. Then there exists a bispecial factor v such that $w = \varphi(v)0^q$ and $M(w) = \varphi(M(v))$.*

Using Observation 6, it suffices to consider bispecial factors of u_β that do not contain 1 to obtain all possible cardinalities of the sets of return words.

Theorem 7. *Let w be a factor of u_β. Then $2 \leq \#M(w) \leq 3$.*

Proof. Let us describe return words of bispecial factors that do not contain 1, i.e., that are equal to $0^r, r \leq p-1$.

1. Let $r \leq q$. Then all the return words of 0^r are the following ones:

$$0 \text{ since there is the block } 0^p \in L(u_\beta),$$

$$0^r 1 \text{ since there occurs the block } 0^p 10^p \in L(u_\beta).$$

2. Let $q < r \leq p-1$. Then all the return words are the following ones:

$$0 \text{ since there is the block } 0^p \in L(u_\beta),$$

$$0^r 1 \text{ since there occurs the block } 0^p 10^p \in L(u_\beta),$$

$$0^r 10^q 1 \text{ since there is the block } 0^p 10^q 10^p \in L(u_\beta).$$

It is apparent that there are no other return words of 0^r. $\qquad\square$

Observation 8. *From the proof of Theorem 7, we can notice that in the case $q = p-1$, $\#M(w) = 2$ for all the factors of u_β. This confirms that u_β is Sturmian in this case.*

References

[1] P. Ambrož, C. Frougny, Z. Masáková, E. Pelantová, *Palindromic complexity of infinite words associated with simple Parry numbers*, to appear in Annales de l'Institut Fourier (2005), pp. 25.

[2] Ľ. Balková, *Complexity for infinite words associated with quadratic non-simple Parry numbers*, submitted to Journal of Geometry and Symmetry in Physics (WGMP 2005 Proceedings) (2006).

[3] Ľ. Balková, Z. Masáková, *Palindromic complexity associated with quadratic non-simple Parry numbers,* submitted to RAIRO – J. Theor. Inform. Appl. (2006).

[4] Ľ. Balková, E. Pelantová, O. Turek: *Combinatorial and arithmetical properties of infinite words associated with quadratic non-simple Parry numbers*, submitted to RAIRO – J. Theor. Inform. Appl. (2006).

[5] Č. Burdík, C. Frougny, J.P. Gazeau, R. Krejcar, *β-integers as natural counting systems for quasicrystals.* J. Phys. **A 31** (1998), 6449-6472.

[6] D. Damanik, L.Q. Zamboni, *Combinatorial properties of Arnoux-Rauzy subshifts and applications to Schrödinger operators.* Rev. Math. Phys. **15** (2003), 745-763.

[7] S. Fabre. *Substitutions et β-systèmes de numération.* Theoret. Comput. Sci. **137** (1995), 219–236.

[8] C. Frougny, Z. Masáková, E. Pelantová, *Complexity of infinite words associated with β-expansions*. Theor. Appl. **38** (2004), 163-185.

[9] L.S. Guimond, Z. Masáková, E. Pelantová, *Arithmetics of β-expansions*. Acta Arithmetica 112.1 (2004).

[10] J. Lagarias, *Geometric models for quasicrystals I. Delone sets of finite type*. Discrete Comput. Geom. **21** (1999), 161-191.

[11] Y. Meyer, *Quasicrystals, Diophantine approximation, and algebraic numbers*. In 'Beyond Quasicrystals (Berlin, 1995)', F. Axel and D. Gratias (eds.), *Les éditions de physique*, Springer (1995).

[12] W. Parry, *On the β-expansions of real numbers*. Acta Math. Acad. Sci. Hungar.**11** (1960) 401-416.

[13] M. Queffélec, *Substitution Dynamical Systems – Spectral Analysis*. Lecture Notes in Math. **1294**, Springer, Berlin, (1987).

[14] K. Schmidt, *On periodic expansions of Pisot numbers and Salem numbers*. Bull. London Math. Soc. **12**(1980), 269-278.

[15] W.P. Thurston, *Groups, tilings, and finite state automata*. Geometry supercomputer project research report GCG1, University of Minnesota (1989).

[16] L. Vuillon, *A characterization of Sturmian words by return words*. European Journal of Combinatorics **22** (2001), 263-275.

Numerical Simulation of Air Flow and Pollution Transport in the Atmospheric Boundary Layer

Petr Bauer

3rd year of PGS, email: `bauerp@kmlinux.fjfi.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Zbyněk Jaňour, Institute of Thermomechanics, AS CR

Abstract. We develop a mathematical model based on Navier-Stokes equations for viscous incompressible flow and diffusion-convection equation describing pollution transport, and solve the model using finite element method (FEM). We use a simple algebraic model with turbulent viscosity. The FEM allows us to use different terrain shapes and study their influence. Essential part of the work is numerical analysis, which examines the properties of algorithms with respect to numerical parameters. We present the recent results of flow over a square gap.

Abstrakt. Vyvíjíme matematický model založený na Navier-Stokesových rovnicích pro nestlačitelné vazké proudění a difuzně-konvekční rovnici popisující transport znečištění, a řešíme jej metodou konečných prvků. Používáme jednoduchý algebraický model turbulence s turbulentní viskozitou. Metoda konečných prvků nám umožňuje zkoumat vliv různých tvarů terénu. Důležitou částí práce je numerická analýza zkoumající chování algoritmů v závislosti na numerických parametrech. Na závěr uvádíme nedávno získané výsledky proudění přes čtvercovou jámu.

1 Introduction

We use a 2D model of air flow and pollution transport on a polygonal domain Ω which represents a vertical cut through landscape. We consider the case of stationary Navier-Stokes flow and diffusion-convection equation for one type of pollutant. We solve the following system of equations on $(0, T) \times \Omega$:

$$
\begin{aligned}
\frac{\partial c(t, x)}{\partial t} + \vec{v}(x)\operatorname{grad} c(t, x) &= D \triangle c(t, x) + f(t, x) \\
\vec{v}(x)\nabla \vec{v}(x) - \nu \triangle \vec{v}(x) + \operatorname{grad} p(x) &= \vec{g}(x) \\
\operatorname{div} \vec{v}(x) &= 0
\end{aligned}
$$

$$
\begin{aligned}
c(0, x) &= c_0(x) \quad x \in \Omega \\
\frac{\partial c}{\partial \vec{n}}\big|_{terr} &= 0 \\
\vec{v}\big|_{terr} &= \vec{v}_t
\end{aligned}
$$

where $\Omega \subset R^2$ is a bounded domain derived from a rectangle by substitution of the bottom edge by a piecewise linear line representing the terrain, $c(t, x)$ is the concentration of pollutant, $\vec{v}(x)$ is the velocity, c_0 is the initial condition for concentration, $\vec{n}$ is the unit outer normal and *terr* denotes the terrain.

On the surface, we are using Neumann boundary condition for concentration and Dirichlet boundary condition for velocity, which is appropriate from the physical point of view; see [9]. We can use either Dirichlet or Neumann boundary conditions on the other parts of the boundary, like $c\mid_{in}= c_i$, $\vec{v}\mid_{in}= \vec{v}_i$, $\frac{\partial c}{\partial \vec{n}}\mid_{out}= 0$, $\frac{\partial \vec{v}}{\partial \vec{n}}\mid_{out}= 0$. The term $f(t,x)$ represents the pollution source and $\vec{g}(x)$ is the external force.

2   Weak formulation of Navier-Stokes problem

Let $V = (\mathring{W}_2^{(1)}(\Omega))^2$, $X = (W_2^{(1)}(\Omega))^2$, $H = \{q \in L^2(\Omega) : \int_\Omega q\,dx = 0\}$, $\vec{w} \in X$: $\vec{w}\mid_{\partial\Omega}= \vec{v}_\Gamma$ in weak sense, $\int_{\partial\Omega} \vec{w}\vec{n}\,dS = \int_\Omega \operatorname{div}\vec{u}\,dx = 0$ for Dirichlet boundary condition. We denote $\vec{u} = \vec{v} - \vec{w}$

$$((\vec{w},\vec{s})) = \int_\Omega \sum_{i,j=1}^2 \frac{\partial w_i}{\partial x_j}\frac{\partial s_i}{\partial x_j} = (\nabla\vec{w}, \nabla\vec{s}), \quad b(\vec{u},\vec{v},\vec{s}) = \frac{1}{2}\int_\Omega \sum_{i,j=1}^2 (u_j\frac{\partial v_i}{\partial x_j}s_i - u_j v_i\frac{\partial s_i}{\partial x_j})$$

We seek $v \in X$ and $p \in H$, such that:

$$\begin{aligned}
((\vec{v},\vec{s})) + b(\vec{v},\vec{v},\vec{s}) - (p, \operatorname{div}\vec{s}) &= (\vec{g},\vec{s}) - ((\vec{w},\vec{s})) &&\forall \vec{s} \in V \\
(q, \operatorname{div}\vec{u}) &= -(q, \operatorname{div}\vec{w}) &&\forall q \in H
\end{aligned}$$

Index h denotes finite-dimensional subspaces $V^h \subset V$, $X^h \subset X$, $H^h \subset H$. The mixed formulation in finite-dimensional case stands:

$$\begin{aligned}
((\vec{v}^h,\vec{s}))_h + b_h(\vec{v}^h,\vec{v}^h,\vec{s}^h) - (p^h, \operatorname{div}_h\vec{s})_h &= (\vec{g}^h,\vec{s})_h - ((\vec{w}^h,\vec{s}))_h \\
(q, \operatorname{div}_h\vec{u}^h)_h &= -(q, \operatorname{div}_h\vec{w}^h)_h &&\forall \vec{s} \in V^h \quad \forall q \in H^h
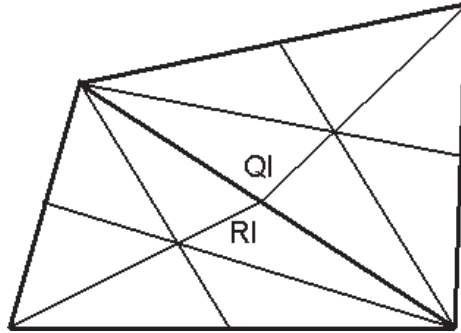\end{aligned}$$

Figure 1: Lumped regions

The nonlinear term $b_h(\vec{u}^h, \vec{u}^h, \vec{s}^h)$ must be computed iteratively. Direct application of this approach gives solution with oscillations. We are using the upwinding technique proposed by [6], which creates the dual triangulation given by the barycentric nodes of the original triangulation. Thus, we obtain the lumped regions R_l around each mid-side node Q_l (Fig. 1). The basic idea of this approach is splitting the domain into the lumped regions and rewriting the nonlinear term in the following form:

$$
\begin{aligned}
b(\vec{u}, \vec{v}, \vec{w}) &= \int_\Omega \left(u_1 \frac{\partial v_1}{\partial x_1} + u_2 \frac{\partial v_1}{\partial x_2} \right) w_1 + \left(u_1 \frac{\partial v_2}{\partial x_1} + u_2 \frac{\partial v_2}{\partial x_2} \right) w_2 = \\
&= \int_\Omega \left(\frac{\partial(u_1 v_1)}{\partial x_1} + \frac{\partial(u_2 v_1)}{\partial x_2} \right) w_1 + \left(\frac{\partial(u_1 v_2)}{\partial x_1} + \frac{\partial(u_2 v_2)}{\partial x_2} \right) w_2 - \\
&\quad - v_1 w_1 \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) - v_2 w_2 \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) = \\
&= \sum_l \int_{R_l} \left(\frac{\partial(u_1 v_1)}{\partial x_1} + \frac{\partial(u_2 v_1)}{\partial x_2} \right) w_1 + \sum_l \int_{R_l} \left(\frac{\partial(u_1 v_2)}{\partial x_1} + \frac{\partial(u_2 v_2)}{\partial x_2} \right) w_2 - \\
&\quad - \sum_l \int_{R_l} v_1 w_1 \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) - \sum_l \int_{R_l} v_2 w_2 \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right)
\end{aligned}
$$

After numerical integration and application of Green Theorem:

$$
\begin{aligned}
b(\vec{u}, \vec{v}, \vec{w}) &\approx w_1(Q_l) \int_{R_l} \mathrm{div}(\vec{u} v_1) + w_2(Q_l) \int_{R_l} \mathrm{div}(\vec{u} v_2) - \\
&\quad - v_1 w_1(Q_l) \int_{R_l} \mathrm{div}\,\vec{u} - v_2 w_2(Q_l) \int_{R_l} \mathrm{div}\,\vec{u} = \\
&= w_1(Q_l) \int_{\partial R_l} v_1 \vec{u}\vec{n}\,dS + w_2(Q_l) \int_{\partial R_l} v_2 \vec{u}\vec{n}\,dS - \\
&\quad - v_1(Q_l) w_1(Q_l) \int_{\partial R_l} \vec{u}\vec{n}\,dS + v_2(Q_l) w_2(Q_l) \int_{\partial R_l} \vec{u}\vec{n}\,dS
\end{aligned}
$$

The term $\int_{\partial R_l} v_i \vec{u}\vec{n}\,dS$ is now approximated by $\sum_k v_i(Q_{\tilde{k}}) \int_{r_{lk}} \vec{u}\vec{n}\,dS$ where k goes through the neighbouring mid-side nodes to Q_l and r_{lk} refers to the part of boundary of the lumped region R_l between nodes Q_l and Q_k. Then $v_i(Q_{\tilde{k}})$ is a combination of values of v_i in these two nodes according to the flux through r_{lk}. This approach leads to the iterative scheme:

$$
\begin{pmatrix} A(u^k) & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u^k \\ -p^k \end{pmatrix} = \begin{pmatrix} G(u^k) \\ H \end{pmatrix} \tag{1}
$$

3 Weak Formulation of Diffusion-Convection Equation

We use implicit Rothe method [4] and the method of characteristics [5] which separates diffusion from convection. This approach uses pre-computed velocity field and is therefore independent on type of the flow used. Application of these methods leads to linear system

with positive-definite matrix; see [10]. For $i = 1, \ldots, m$:

$$\sum_{j=1}^{m} \alpha_j^k [(v_j, v_i) + \tau D(\nabla v_j, \nabla v_i)] = (\tau f^k + c^{k-1} \circ \varphi^k, v_i) - \tau D(\nabla w, \nabla v_i)$$

4 Algebraic Turbulence Model

The algebraic turbulence model is a very simple way of simulating some turbulent properties of the atmospheric boundary layer. We introduce the turbulent viscosity ν_t as the function of height, instead of using the constant viscosity coefficient. The actual dependence is given by following formula:

$$\nu_t = \frac{\kappa^2 \mu^2 u_*^2}{f} K_M \quad \mu = 1.0 \quad f = 10^{-4} \quad \kappa = 0.4 \quad u_* = 0.1$$

$$
\begin{aligned}
K_M &= \mu Z \quad\quad Z < h \quad \delta = 10^3 \quad h = 0.1\delta \quad Z = z/h \\
&= \mu h \quad\quad Z \geq h
\end{aligned}
$$

where δ is the thickness of the boundary layer, u_* is the friction velocity, K_M is dimensionless turbulent exchange coefficient and Z is the relative height. The values of constants are given by physical experiments.

The value of ν_t tends to zero at the proximity of the surface, which is not acceptable. The boundary condition must therefore be set in some nonzero height z_1 which corresponds to terrain roughness. For the inner city model, we consider the values $z_1 = 20m$ and $u(z_1) = 7.5m/s$. The velocity profile is given by $u(z) = u(z_1)(z/z_1)^{0.28}$ to match the velocity of geostrophic flow on the top boundary.

5 Numerical Solution using the Finite Element Method

We have chosen the finite element method, in order to treat different terrain shapes easily. We use linear Lagrange elements for concentration, Cruzeix-Raviart (Fig. 2) elements for velocity and piecewise constant elements for pressure.
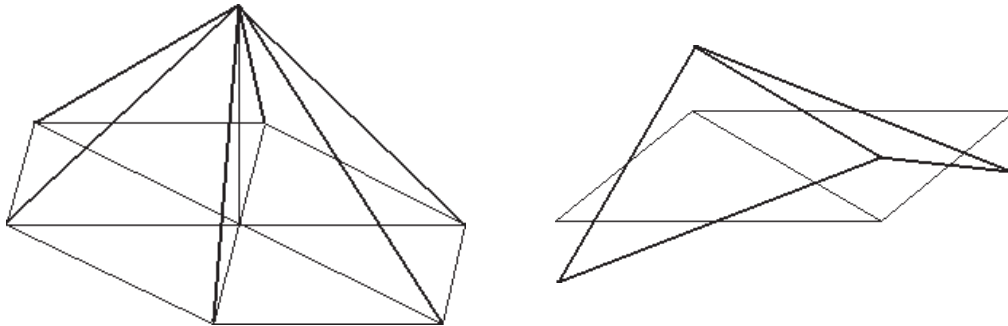

Figure 2: Lagrange and Cruzeix-Raviart elements

Current mesh structure allows storing of multiple meshes together with their hierarchic structure to support the more efficient multigrid solver, which is being developed. Meanwhile, we use Uzawa iterations and BiCG method with simple diagonal preconditioning to solve system (1). We can treat two different types of boundary conditions for velocity on each part of the boundary: Dirichlet and do-nothing.
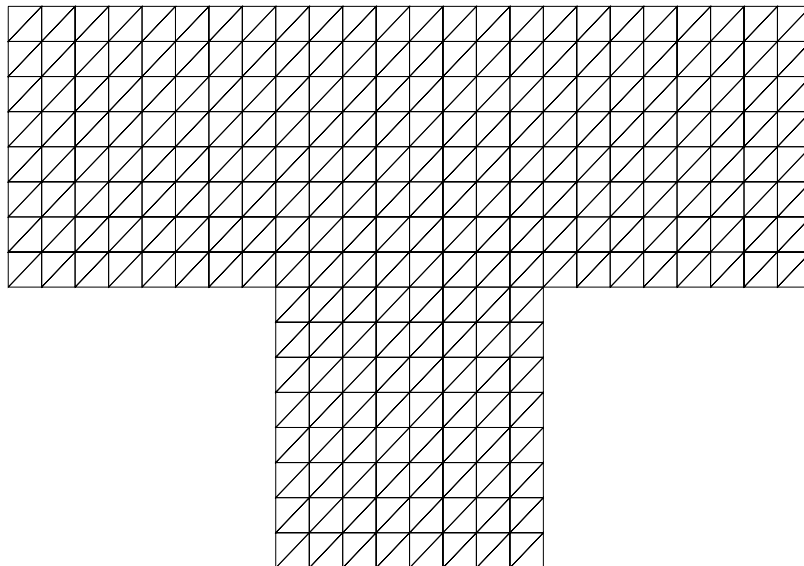

Figure 3: Triangulation of the domain

6   Results

On the last pages, we present recent results of steady state Navier-Stokes flow over a unit square gap in the case of Reynolds number $Re = 1000$ with parabolic velocity profile on the inlet. The velocity and pressure fields are shown. Note, that the pressure is determined up to constant; we have chosen zero as a mean value in this case. For comparison, we also show the results of stationary Stokes flow with the same boundary condition.

7   Conclusion

We develop a mathematical model of pollution transport in the ABL based on Navier Stokes equations and diffusion-convection equation, and solve it using the finite element method. The model includes a simple algebraic model of turbulence and allows us to prescribe different types of boundary conditions on each part of the boundary. We have shown the numerical results of flow over a square gap for comparison with reference data.

The immediate goals are implementation of multigrid solver and extension of the model into 3D. Next step will be the implementation of an advanced turbulence model and comparison of its results with experimental data provided by the Institute of Thermomechanics of the Academy of Sciences of the Czech Republic.

Simultaneously, we are trying to parallelize the code in order to solve complicated 3D problems. At this occasion, I would like to express my thanks to the members of HPC Europa project, for their aid and hospitality during my summer stay in research center CINECA in Bologna, where a significant amount of work on this topic was done.

References

[1] P. G. Ciarlet, *The finite-element method for elliptic problems,* North-Holland, Amsterdam (1978)

[2] F. Brezzi, M. Fortin, *Mixed and hybrid finite-element methods,* Springer Verlag, New York (1991)

[3] M. Feistauer, *Mathematical methods in fluid dynamics,* Longman, New York (1993)

[4] J. Kačur, *Method of Rothe in evolution equations,* BSB B. G. Teubner Verlagsgesellschaft, Leipzig (1985)

[5] J. Kačur, *Solution of Degenerate Convection-Diffusion Problems by the Method of Characteristics,* SIAM Journal on Numerical Analysis, 39 (2001), 858–879

[6] F. Schieweck, L. Tobiska, *An optimal order error estimate for upwind discretization of the Navier-Stokes equation,* Numerical methods in partial differential equations y.12 n.4 (1996), 407–421

[7] M. Beneš, Z. Jaňour and F. Rys, *Numerical Solution of the Falkner-Skan Equation,* proceedings on 3rd Seminar on Euler and Navier-Stokes Equations (Theory, Numerical Solution, Applications), Institute of Thermomechanics, Prague, May (1998), 13–14

[8] M. Beneš, Z. Jaňour, *A numerical solution of the planetary-boundary-layer equations,* extended abstract in Workshop on urban boundary layer parametrisations, COST Action 715, Zurich, May (2001), EC 2002, 63–72

[9] M. Beneš, R.F. Holub, *Aerosol Wall Deposition in Enclosures Investigated by Means of a Stagnant Layer,* Environment International 22, Suppl. 1 (1996), 883–889

[10] P. Bauer, *Mathematical Modelling and Numerical Simulation of Pollution Transport in the Atmospheric Boundary Layer,* Proceedings on the Conference Topical Problems of Fluid Mechanics, Praha (2005), 7–10
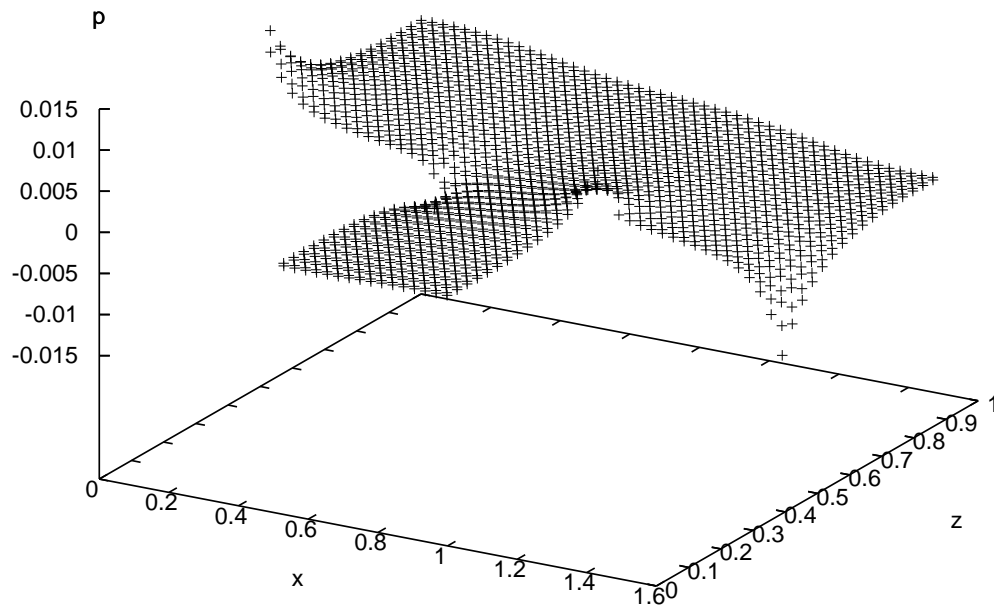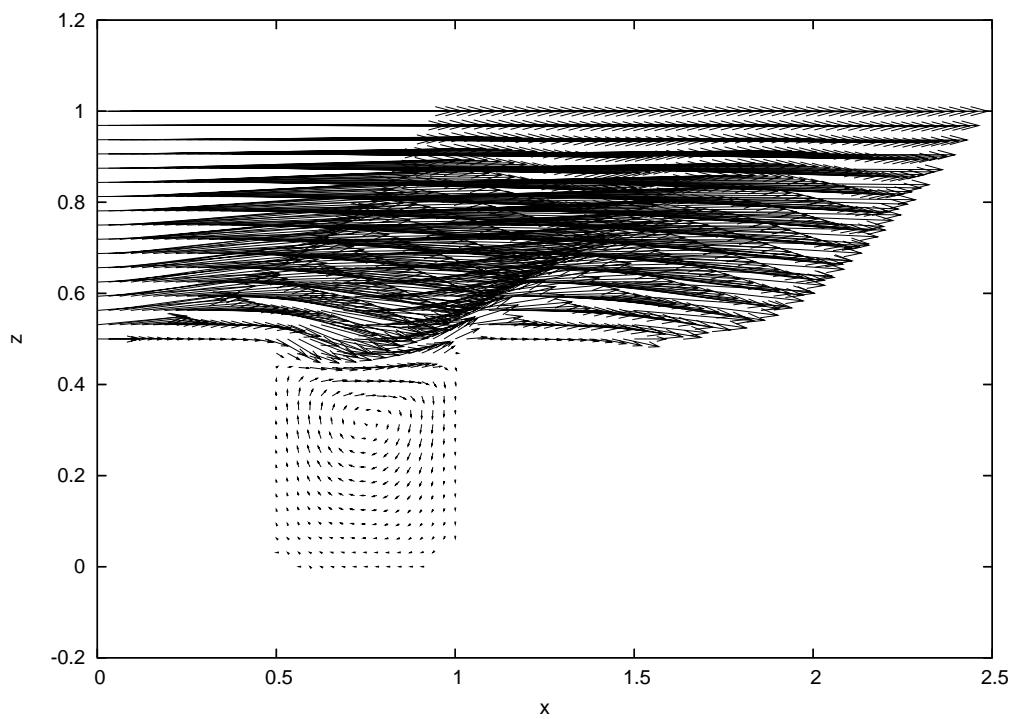
Figure 4: Stokes flow - pressure
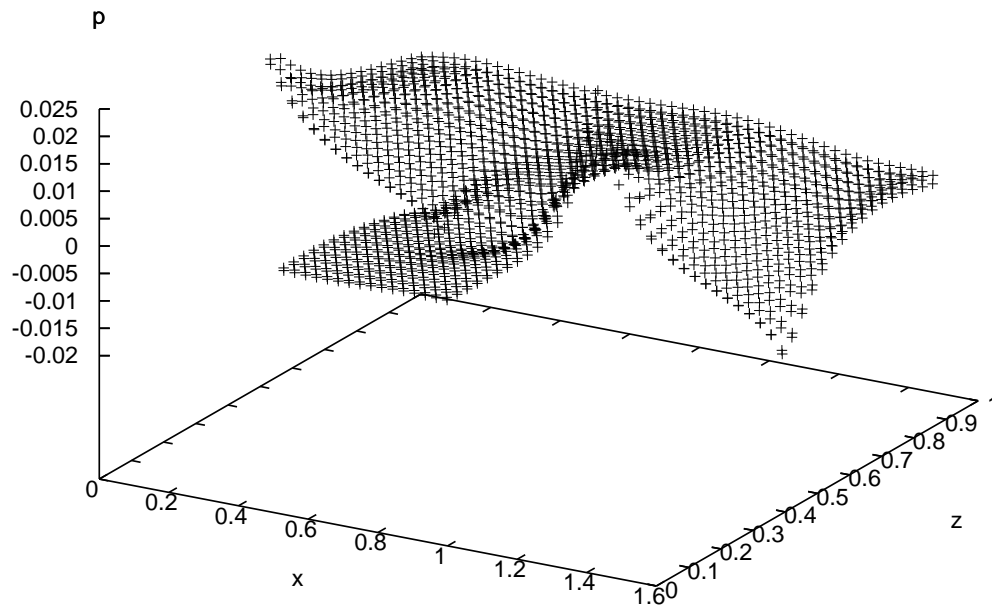

Figure 5: Stokes flow - velocity

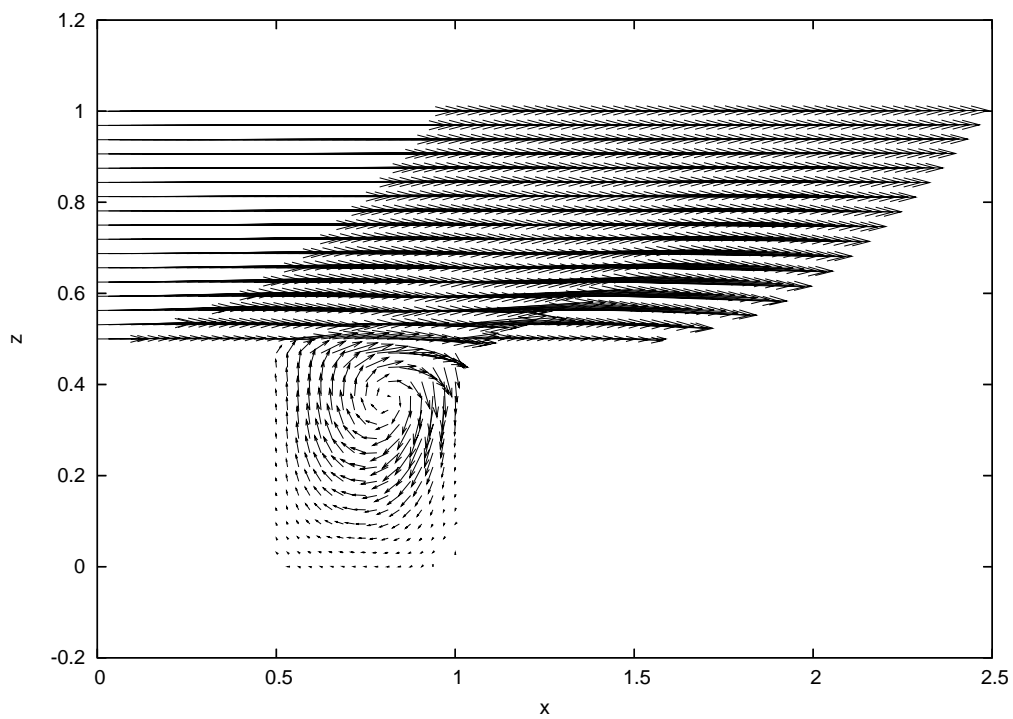Figure 6: Navier-Stokes flow - pressure

Figure 7: Navier-Stokes flow - velocity

Prototype Implementation of the Execution Engine VPU Core[*]

Martin Dráb

3rd year of PGS, email: `drab@kepler.fjfi.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Ladislav Kalvoda, Department of Solid State Engineering, Faculty
of Nuclear Sciences and Physical Engineering, CTU

Abstract. The goal of the project INDECS is to create a flexible software basis for driving an experiment on a neutron diffractometer. The step of creating the first prototype of the Execution Engine (EE) Virtual Processing Unit (VPU) core, which is a key component of the developed experiment control system, had to be done in order to be able to proceed further with other components and finally assembling them into the working system. This article tries to also describe some deviations from the original design draft of the EE VPU, that were considered necessary to do on the path from the theoretical design to the actual implementation.

Abstrakt. Cílem projektu INDECS je vytvořit flexibilní software pro řízení exprimentu na neutronovém difraktometru. Bylo nutné provést krok, ve kterém byl vytvořen první prototyp virtuálního procesoru (VPU) zvaného Execution Engine (EE), který je klíčovou komponentou vyvíjeného systému pro řízení experimentu, aby bylo možné pokračovat s vývojem dalších komponent systému a ve finále z těchto komponent sestavit fungující systém. Tento článek se mimo jiné snaží popsat některé odchylky od originálního návrhu VPU EE, které se při cestě od teoretického návrhu k praktické implementaci ukázaly jako nutné.

1 Introduction

Project INDECS is oriented on designing a flexible software basis for an effective control of the diffractometer experiments, however it should be easily modifiable for a different type of experimental devices. The software should be able to control the moves of the diffractometer as well as to perform the actual data acquisition and analysis. Ability to communicate over the network with data and command transfers, and thus, in fact having the ability of remote controlling or monitoring of the experiment, was requested.

The result is achieved by a series of connected Virtual Processing Units (VPU) called the Execution Engine (EE) which allows all the capability that is required for the system to have. These VPUs have a capability to execute general virtual instructions that can be used to control the data streams passing through them or to perform some basic tasks that can be further supported by a variable instruction set and external modules called the External Execution Modules (EEMs) that can execute complex and possibly target-specific virtual instructions. In order to withstand communication loss, it also has the capability to locally store and execute instruction sequences. And it can do a complex routing of instruction streams.
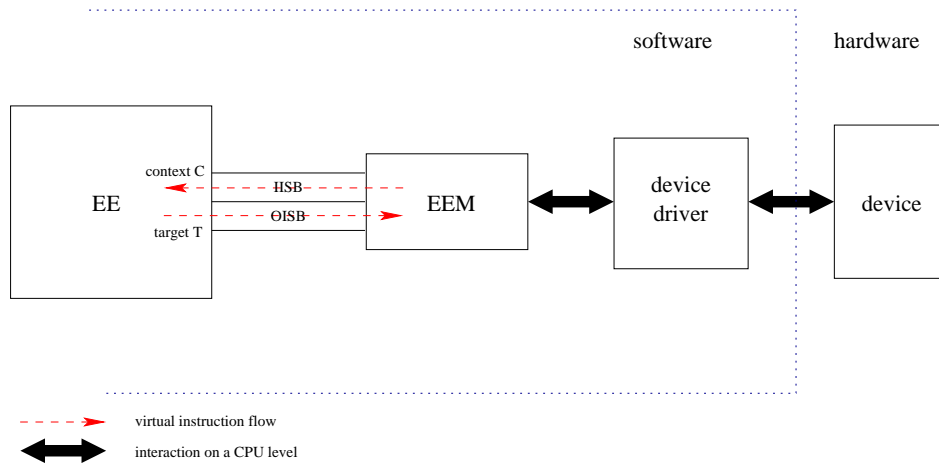

Figure 1: A simple schematics of an EEM application to drive some peace of device.

Before this concept can be extended to a fully working complex system able to drive the diffraction experiment, a first prototype of the actual Execution Engine VPU Core itself had to be created, since it is the key component to the whole system. Another reason for this step is the fact, that some other components are just a derivative of the EE. In this case we had the External Execution Modules in mind, which is in fact to beb implemented as an Execution Engine itself with reduced communication and stream routing capability and without the capability to execute the general basic instruction set common to all EEs, and instead just implement the one device- or function-specific instruction set of possibly quite complex and extended functionality. A simple schematics of an EEM application to drive some peace of device can be seen on Figure 1.

2   Concept

Conceptually the construction of the Execution Engine VPU is based on three layers folded on one another. In order to get to this structure of implementing the EE, we had to realize the fact that if you take a global view on the whole system which we are trying to build here, we can clearly see that in fact it is all about communication between its individual parts. Actually about the instruction and therefore data stream routing and exchange.

The three layers are trying to reflect this. The lowest layer is called a Stream Cache (SC) layer and it deals with a smart buffering that is used as a basic mean of data communication between individual components. The middle layer is called a MultiMedia Stream Routing (MMSR) layer, it is based upon the Stream Cache layer, and it defines separate entities that can be bound together using data streams to exchange information that these entities can work with. The third layer hovering on both of these is the actual implementation of the Execution Engine VPU core itself.

2.1   The MultiMedia Stream Routing Layer

The basic data stream routing is done using so called Transcoders. A Transcoder is a software construct (or, say, an object) that has a variable and theoretically unlimited number of input and output data streams and two functions (or methods) called `scan_resources()` and `transcode()`. The first function is supposed to scan input streams and determine, whether there is enough data available to do one transcoding step, and the second function performs the transcoding step, which could more or less be any mathematical transform transforming some data from the input streams into some other data that can be sent to any of the output streams.

Execution Engines and External Execution Modules are implemented as a special case of these Transcoders with special `scan_resources()` and `transcode()` functions, that process the incoming commands. Execution Engine is also capable of executing locally stored commands which could be stored within and their references are hidden within its data structures.

2.2   The Stream Cache Layer

The data streams that connect multiple Transcoders (and in fact also multiple EEs and EEMs) together are created by the second and bottom-most sublayer that is called a Stream Cache layer. Again, it is a software construct. This time meant to maintain a cache of streaming data, allowing easy access to the data in possibly variable amounts and with little overhead.

The same way as Transcoders can also be used (and are used) for other purposes than making an EE or an EEM, the Stream Cache can be (and is) used for other purposes than just to connect more Transcoders.

The thing is that basically someone has to feed the cache with some data at one end and someone else has to retrieve the data on the other end of the cache. Both of which can be done manually, in which case the cache operates in so called Push-Pull mode. That is the mode that is probably used the most, also for connecting the Transcoders.

Then the cache can also operate in the so called Pull-Automatic or Push-Automatic mode. First of which has a special function attached to the cache, that is automatically called, when the cache is supposed to be fed with data (this can be used for instance for automatic reading of data from files or other data sources). The other mode is its exact opposite, it calls a different attached function whenever the cache needs to release some data (this can be used e.g. for writing data to files).

Data is read from the cache by first blocking a necessary amount within the cache (making sure that the data would not be moved in between by the automatic cache handling mechanism), then processing them in-place and finally releasing them out from the cache. Of course there is also the possibility to just unblock the data without releasing them. This way a function can test the data and if it is not (yet) interested in them, it just lets them there either for someone else or for later time, when there is more data.

A smart mechanism can be allowed to move certain amount of the data from the end of the cache to the beginning, and though creating a feeling of data continuity to the reader. It can even dynamically change the size of the cache buffer when large amounts of data are necessary to be available at one time, that are bigger than the cache itself. Of
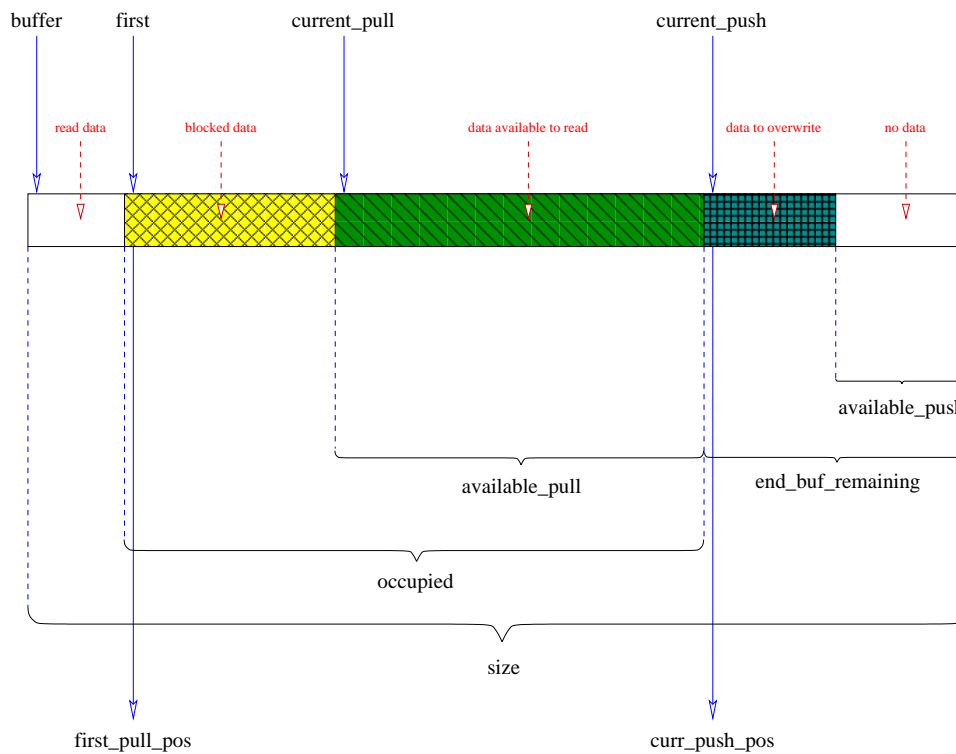
Figure 2: The scheme of the Stream Cache buffer operation.

course these features can be all individually disabled and possibly restricted to certain bounds.

The whole idea behind this was to create a caching mechanism, that from the side of the reader would seam like a continuous cache, i.e., it allways gets the amount it asks for in one peace (as long as the cache's size and some other parameters allow it), that would allow the reader to work with the data directly from the cache, i.e., without a need to do a local copy of the data to process them, and with as little overhead as possible.

3   Implementation

The actual implementation of the third topmost layer, the Execution Engine itself, is mostly following the design draft as stated in [1], chapter 3 and 4. The actual instance of the EE is reslized as a structure called `indecs_ee_t`. Among other things it holds a list of contexts, a list of allocated Local Memory Blocks, a structure with the global registers of the EE, the table of targets that are to be considered as a source of the local routing paths, list of input and output instruction streams (connections) and other things important to the fast and possibly simple operation of the VPU.

3.1 Contexts

Contexts are defined as a concurrent (though cooperating) sources of instructions. They are instatnitated by a structure called `indecs_context_t`. These structures are organized in several ways. The first way is the list of contexts available to the particular instance of the EE. It is a dynamically linked list with a header placed in the `indecs_ee_t` structure and its purpose is to just have all the available contexts listed on one place, so that it is easy to handle them globally, e.g., terminating them properly upon the termination of the EE and such.

The other way of organizing the `indecs_context_t` structures is of more practical use and is aimed towards speeding up the process of the EE's scheduler to be able to faster decide which context is to be chosen to execute the next virtual instruction. According to [1], chapter 3.2, the whole scheduling scheme of the EE is based on a priority based cooperative multitasking. Which means that each process is assigned an integer number called priority. The next context to be scheduled for executing the next virtual instruction is allways the one, that has the biggest priority number and is ready to execute an instruction, i.e., it has a whole instruction that is to be executed next in within the context available.

To speed up the process of searching the the appropriate context the `indecs_context_t` structures are also organized in another way. There is a dynamic list of occupied priorities, i.e., each priprity that contains at least one context has an entry in this list. This list is descendently ordered according to the priority. This allows quickly finding the topmost priority context(s) (it takes a constant time) and possibly quicky skip to the next lower priority which also takes a constatnt time) and so on. Each entry within this list contains another dynamic list of all contexts (i.e., their `indecs_context_t` structures) that are currently on that particular priority level. This organization allows us to easily implement the round-robin principle of executing contexts that are on the same (top) priority, according to [1]. The next context to be scheduled on the particular prioriry level is the one that is on the top of the list connected to that priority within the list of occupied priorities. Upon the scheduling the context is moved to the bottom of that list (which is again a constant time operation), which assures the requested round-robin behaviour.

To speed up the searching of appropriate context based on its Context Identification (CID, see [1]), the list of all contexts is hashed according to the CID in addition to the organization in the dynamic list. To speed up the process of searching the lists particular priority level during the operation of reassigning a context's priority, the list of occupied priority levels is hashed as well, this time using the priority level as a hashing key.

3.2 Instruction Sources

According to [1] a source of instructions can be either an Incomming Isntruction Streaming FIFO Buffer (IISB), which is implemented by an input stream to the Transcoder (that instantiates the EE). This source is only streamable and is not seekable, which means, that it can only execute instructions one after another in the order in which they are stuffed into the IISB. So there are no jump or branch instructions that could explicitly change the order of the execution of the instructions.

The other source of instructions is the Local Memory Block (LMB), which is a dynamically preallocated area of memory that can be filled with instructions and executed. Instructions executed from within the LMB can implement jumps and branching in the execution targetting instructions that are also in an LMB.

While each IISB does have its own context, and so is a virtual process on its own, it is not mandatory for the LMB source to have it as well. An LMB can be either launched as a context on its own and then behave simmilarly to the IISB with jumping and branching, or it can be just loaded with code and its peaces can be called from other contexts as subroutines by the `call` or `ccall` instructions. And this can be done from both contexts based on LMBs and contexts based on IISBs, which is a way how even an IISB based context can implement sort of a jump or branch, even though in such an indirect way. But to implement subroutine calls, each context has to have a special LMB that implements the data structure called stack, which is created automatically upon the creation of each process and its referrence (LMBID) is stored within a special context local register called `SLR` (Stack LMB Referrence). This structure is mainly used to save the return pointer for the next instruction to be called upon the return from the subroutine call, i.e., effectively it saves the CVIP (Current Virtual Instruction Pointer) register (see [1]). As for contexts executing the instruction stream from an IISB, this register allways has a special value of 0, the concept of subroutine calling is also usable on such contexts without a problem.

LMBs are again instantiated by a structure, this time called `indecs_lmb_t`, and these are again linked in a dynamic list with the head of the list grounded within the main EE structure `indecs_ee_t`. Similarly to the list of contexts this list is also hased by the LMB Identification number (LMBID), in order to speed up the process of searching the particular LMB, since the explicit referrence to the data or instructions within an LMB are ususaly done by specifying the LMBID and an offset within the LMB.

3.3   Routing

Again, the routing mechanism is implemented according to [1], chapter 3.10. However in order to use all of the possibble combinations of the routing registers GRENi, GRTEi and GRTNi (you can see the local routing principle using these registers depicted on Figure 3) and to speed things up a bit, another set of 255 1-bit registers called GREi (Global Routing Enable) was added. These registers when set to a 0 disable the operation of the particular GREN, GRTE and GRTN registers with the same index, when set to a 1, these are enabled.

To speed up the process of checking which of the 256 targets the routing should apply to, these new 256 1-bit GRE registers are stored as 8 32-bit values which can be tested for non-zero value as a whole and then possibly dig deeper into the individual bits, if it is going to be necessary. Another thing that should speed up this process is a special array of 256 entities held within the `indecs_ee_t` structure. This array contains for each of the targets a reference counter, i.e., the number of local routing paths that take this particular target as a source for the routing and though are of interest to the routing mechanism, and an index of the first local routing path (by a local routing path in this context we understand the three registers GRENi, GRTEi and GRTNi with the same index i), i.e., the one with the lowest index, that takes this target as a source. The
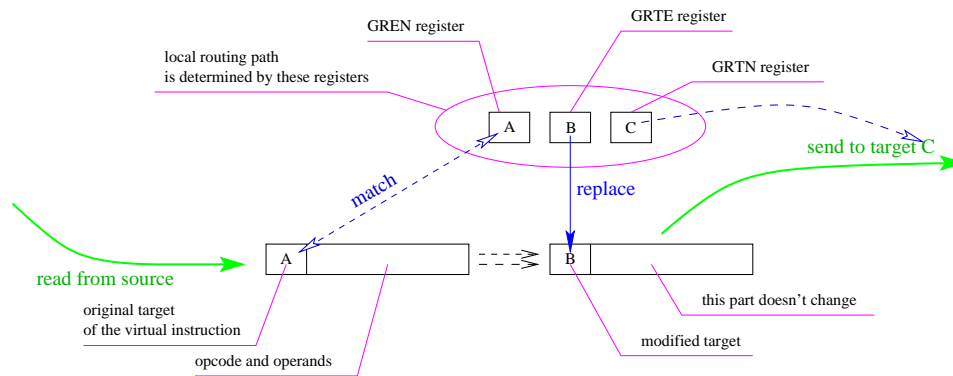
Figure 3: The scheme of local routing using a GREN, GRTE and GRTN registers.

searching of the local routing paths for the specific target is then eased in a way, that we explicitly know which is the first one (its index) and that we know how many of such local routing paths there is, so we don't need to go through all of the 256 GREN registers for each executed instruction, in order to see where we need to route it.

4   Conclusion

The first prototype of the Execution Engine VPU core has been constructed. It yet needs to undergo a series of testing and debugging in order to optimize and determine its characteristics. However the process of integrating it into a full experiment control system still requires a lot of work. First the derivative of the EE, the External Execution Module (EEM) prototype has to be constructed. Next step would be a construction of a simple network consisting of several of these components interconnected together and testing of its operability. Further a series of special EEMs to interact with the actual hardware of the experiment (in this case devices belonging to the KSN-2 diffractometer, whose driving is the ultimate goal of this project) have to be created and tested. Another task would be to create a simple and flexible user interface to let the user actually interact with the experiment control system.

References

[1] M. Dráb. *Project INDECS: The Design of a Superior System of Data Collection and Analysis*. (Diploma Thesis) Faculty of Nuclear Science and Physical Engineering, Czech Technical University, Prague, (2003).

Minimum Information Loss Cluster Analysis for Categorical Data*

Jan Hora

2nd year of PGS, email: `hora@utia.cas.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Jiří Grim, Institute of Information Theory and Automation of the Czech Academy of Sciences

Abstract. The EM algorithm has been used repeatedly to identify latent classes in categorical data by estimating finite distribution mixtures of product components. Unfortunately, the underlying mixtures are not uniquely identifiable and moreover, the estimated mixture parameters are starting-point dependent. For this reason we assume the latent class model only to define a set of elementary properties and the related statistical decision problem. In order to avoid the problem of unique identification of latent classes we propose a hierarchical "bottom up" cluster analysis based on unifying the latent classes sequentially. The clustering procedure is controlled by minimum information loss criterion.

Abstrakt. Shluková analýza kategoriálních dat je obecně obtížnou úlohou, ve které není možné využít standardních postupů, protože u kategoriálních není jednoduše definovatelná obecně platná metrika. Jisté možnosti v tomto směru nabízí hledání tzv. latentních tříd, pomocí EM algoritmu, které je však velmi omezeno. V prvé řadě tím, že shluky hledá pouze ve tvaru komponent směsi, a v druhé naráží na teoretický problém neidentifikovatelnosti směsi a její závislost na počátečním řešení. V našem případě takto nalezené třídy pokládáme pouze za referenční body, které definují transformaci kategoriálních dat do reálného prostoru. Po transformaci je využito hierarchické shlukování s využitím kritéria Informační ztráty.

1 Introduction

The concept of cluster analysis is closely related to the similarity of objects or distance of data vectors defined by a metric. The cluster analysis of categorical (nominal, qualitative) data is difficult because the standard arithmetical operations are undefined and also there is no generally acceptable definition of distance for multivariate categorical data. For these and other reasons the standard methods of cluster analysis cannot be applied directly to multivariate categorical data.

One of the first ideas of statistical multivariate analysis of categorical data appears to have originated with Lazarsfeld [22]. In the framework of sociological research he proposed the fitting of multivariate Bernoulli mixtures to binary data with the aim to identify possible latent classes of respondents. Serious drawback of the Lazarsfeld's idea has been the tedious and somewhat arbitrary methods used for fitting the models. The numerical

---

problems have been removed only by the computationally efficient EM algorithm [6]. Discussion of the latent class analysis from a statistical point of view can be found in Goodman [9] and Fielding [7]. Other approaches to latent structure- and latent variable models are discussed by Bartholomew [2], (see also [8], [23]). In the last years the original idea of Lazarsfeld has been widely applied and frequently modified by different authors (cf. e.g. [19], [28]).

In this paper we propose a new approach to cluster analysis of categorical data in the context of data mining. Applying the latent class model to large multivariate databases we have to assume large number of classes ($M \approx 10^1 \div 10^2$) and multiple solutions of comparable quality. In order to avoid the difficulty of unique identification we propose a hierarchical clustering procedure based on sequential unifying the "elementary" latent classes. The procedure is controlled by the minimum information loss criterion.

2   Latent Class Model

Let us suppose that some objects are described by a vector of discrete variables taking values from finite sets:

$$\boldsymbol{x} = (x_1, \ldots, x_N), \quad x_n \in \mathcal{X}_n, \; |\mathcal{X}_n| < \infty, \quad \boldsymbol{x} \in \mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_N. \tag{1}$$

We assume that the variables are categorical (i.e. non-numerical, nominal, qualitative) without any type of ordering. Considering the problem of cluster analysis we are usually given a set of data vectors

$$\mathcal{S} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(K)}\}, \quad \boldsymbol{x}^{(k)} \in \mathcal{X}. \tag{2}$$

The goal of cluster analysis is to partition the set $\mathcal{S}$ into "natural" well separated subsets which correspond to similar objects

$$\Re = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_M\}, \quad \mathcal{S} = \cup_{j=1}^{J} \mathcal{S}_j, \quad \mathcal{S}_i \cap \mathcal{S}_j = \emptyset, \; \text{ for } i \neq j. \tag{3}$$

The concept of cluster analysis is closely related to some similarity or dissimilarity measures. Unfortunately, in case of categorical variables the arithmetical operations are undefined and therefore we cannot compute means and variances nor there is any generally acceptable way to define distance for the categorical data vectors $\boldsymbol{x} \in \mathcal{X}$. For this reason the standard algorithms of cluster analysis are not directly applicable to categorical data.

One of the first statistically justified approaches to clustering discrete data is the latent structure analysis of Lazarsfeld [22], who proposed to identify latent classes in binary data by estimating multivariate Bernoulli mixtures. The method is easily generalized to categorical variables and it is often discussed in recent literature in different modifications as "latent class analysis" [28]. In particular we define the latent class model as finite mixture of product components

$$P(\boldsymbol{x}) = \sum_{m \in \mathcal{M}} w_m F(\boldsymbol{x}|m), \quad \boldsymbol{x} \in \mathcal{X}, \; (w_m > 0), \quad \mathcal{M} = \{1, \ldots, M\}, \tag{4}$$

$$F(\boldsymbol{x}|m) = \prod_{n \in \mathcal{N}} f_n(x_n|m), \quad \mathcal{N} = \{1, \ldots, N\} \tag{5}$$

where the probability w_m is the probabilistic weight of m-th component, $f_n(x_n|m)$ are the conditional (component specific) univariate discrete distributions and $\mathcal{M}, \mathcal{N}$ are the index sets of components and variables respectively.

The formula (4) can be viewed as a model of conditional independence with respect to the index variable m. It is not restrictive in case of discrete variables since any discrete probability distribution can be expressed as a product mixture (4) if the number of components may be chosen large enough (cf. [17]).

The standard way of estimating mixtures is to use EM algorithm (cf. [6], [10], [21], [24]). In particular to compute maximum-likelihood estimates of mixture parameters we maximize the log-likelihood function

$$L = \frac{1}{|\mathcal{S}|} \sum_{x \in \mathcal{S}} \log P(\boldsymbol{x}) = \frac{1}{|\mathcal{S}|} \sum_{x \in \mathcal{S}} \log \left[\sum_{m \in \mathcal{M}} w_m F(\boldsymbol{x}|m) \right] \tag{6}$$

.

3 Non-unique Identification of Latent Classes

It can be seen that the distribution mixture (4) naturally defines a statistical decision problem $\{\mathcal{X}, w_m F(.|m), m \in \mathcal{M}\}$. Having estimated the mixture parameters we can compute the Bayes probability $q(m|\boldsymbol{x})$ that a data vector $\boldsymbol{x} \in \mathcal{S}$ "relates" to the m-th component. Using the Bayes decision function [1]

$$d(\boldsymbol{x}) = \arg \max_{j \in \mathcal{M}} \{q(j|\boldsymbol{x})\}, \quad \boldsymbol{x} \in \mathcal{X}. \tag{7}$$

we can define the "latent class" partition $\Re$ of the set $\mathcal{S}$ by classifying the points $\boldsymbol{x} \in \mathcal{S}$ by means of $d(\boldsymbol{x})$:

$$\Re = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_M\}, \quad \mathcal{S}_m = \{\boldsymbol{x} \in \mathcal{S} : d(\boldsymbol{x}) = m\}, \quad m \in \mathcal{M}. \tag{8}$$

Let us recall that the partition $\Re$ represents the result of latent class analysis in the original form proposed by Lazarsfeld (cf. [22], [28]). The posterior weights $q(m|\boldsymbol{x})$ can be viewed as membership functions of the estimated latent classes. They are particularly useful if there is some interpretation of the mixture components, e.g. if the components correspond to some real "latent classes" [22], "hidden causes" [23] or "clusters" having a specific meaning.

The latent class model (4) seems to be one of the most widely applicable tool of cluster analysis of categorical data. Thus e.g. the original idea of Lazarsfeld has been used by many authors to identify individual classes of bacteria (cf. e.g. [19]) and more recently Vermunt et al. [28] describe different modifications of the latent class analysis as applied in other fields.

A serious disadvantage of the latent class analysis relates to the fact that the resulting clusters may be non-unique. It is obvious that, if the estimated mixture is not defined uniquely, then the corresponding interpretation of data in terms of latent classes may

---

[1]In case of ambiguity we set the decision function $d(\boldsymbol{x})$ equal to the smallest index $j \in \mathcal{M}$ with the maximum property.

become questionable. Unfortunately, there are at least three sources of uncertainty which may influence the resulting form of the estimated mixture. First, there is no exact method to choose the proper number of components (cf. [21]). Another source of multiple solutions is the existence of local maxima of the log-likelihood function. For this reason we can expect different locally optimal solutions depending on the chosen initial parameters. However, even if we succeed to manage the computational aspects of the latent class model identification, there is still the well known theoretical problem that the latent class model is not identifiable (cf. [25], [3], [19]).

In practice the non-identifiability of latent class models does not seem to have serious consequences since the classes can be uniquely identified by external knowledge or by additional constraints (cf. [5], [19]). As it appears, in many practical problems the well defined components can be spontaneously identified [5]. However, in the context of data-mining the problem of non-identifiable latent classes becomes more essential.

In case of large multidimensional real-life databases, which are typical for data-mining problems, the form of the estimated distribution is generally unknown. Obviously, the primary goal is to approximate the unknown distribution instead of exact identification of a specific set of mixture parameters. Multidimensional spaces are "spars" and therefore, in order to achieve reasonable approximation accuracy, we have to choose relatively large number of components ($M \approx 10^2 \div 10^3$). Some of the resulting components usually have very low weights and may be omitted without observable consequences. In this sense the exact number of components M is less relevant. According to our practical experience (cf. [14], [15], [16]) there are usually numerous local maxima of the likelihood function having similar values. Expectably, the corresponding mixture estimates are different but of comparable quality. From the point of view of approximation accuracy the influence of initial parameters is negligible and the EM algorithm can be initialized randomly.

4   Statistical Cluster Decision Problem

We can conclude that in case of data-mining we have to consider multivariate latent class models with relatively large number of classes. There is usually large variability of the estimated parameters which, on the other hand, correspond to similar values of the log-likelihood function and comparable approximation accuracy of the estimated mixture. For this reason the latent classes themselves are not suitable to define directly the latent structure of large multivariate categorical data sets. In this paper we propose a hierarchical "bottom up" clustering procedure which consists in sequential pairwise unifying of the elementary latent classes. A suitable criterion to control the process of hierarchical cluster analysis is the statistical decision information.

As shown in Sec. 3 the identification of classes represents a classification problem which can be solved in terms of statistical decision-making. Considering the probabilistic description of latent classes we can compute the statistical decision information $I(\mathcal{X}, \mathcal{M})$ about $\mathcal{M}$ contained in $\mathcal{X}$. By means of Shannon formula we obtain

$$I(\mathcal{X}, \mathcal{M}) = H(\mathcal{M}) - H(\mathcal{M}|\mathcal{X}), \qquad H(\mathcal{M}) = \sum_{m \in \mathcal{M}} -w_m \log w_m, \qquad (9)$$

$$H(\mathcal{M}|\mathcal{X}) = \sum_{x \in \mathcal{X}} P(\boldsymbol{x}) H_x(\mathcal{M}) = \sum_{x \in \mathcal{X}} P(\boldsymbol{x}) \sum_{m \in \mathcal{M}} -q(m|\boldsymbol{x}) \log q(m|\boldsymbol{x}). \tag{10}$$

Here $H(\mathcal{M})$ is the uncertainty connected with estimating the outcome $m \in \mathcal{M}$ of a random experiment with the probabilities $\{w_1, \ldots, w_M\}$. Given a vector $\boldsymbol{x} \in \mathcal{X}$ we can improve the estimation accuracy by computing the more specific conditional probabilities $q(m|\boldsymbol{x})$. In this way we make use of the decision information contained in the latent class model.

By means of the decision function $d(\boldsymbol{x})$ we define partition Φ of the space $\mathcal{X}$

$$\Phi = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_M\}, \quad \mathcal{X} = \cup_{j=1}^{M} \mathcal{X}_j, \quad \mathcal{X}_i \cap \mathcal{X}_j = \emptyset, \quad \text{for } i \neq j. \tag{11}$$

Using the partition Φ we can simplify the decision-making by considering only the "averaged" conditional probabilities

$$q(m|\mathcal{X}_j) = \frac{w_m F(\mathcal{X}_j|m)}{P(\mathcal{X}_j)} = \sum_{x \in \mathcal{X}_j} \frac{w_m F(\boldsymbol{x}|m)}{P(\mathcal{X}_j)} = \sum_{x \in \mathcal{X}_j} \frac{P(\boldsymbol{x})}{P(\mathcal{X}_j)} q(m|\boldsymbol{x}). \tag{12}$$

The statistical decision information contained in the averaged conditional probabilities $q(m|\mathcal{X}_j)$ can be expressed by Eq.:

$$I(\Phi, \mathcal{M}) = H(\mathcal{M}) - H(\mathcal{M}|\Phi) = H(\mathcal{M}) - \sum_{j \in \mathcal{M}} P(\mathcal{X}_j) H_{\mathcal{X}_j}(\mathcal{M}), \tag{13}$$

$$H_{\mathcal{X}_j}(\mathcal{M}) = \sum_{m \in \mathcal{M}} -q(m|\mathcal{X}_j) \log q(m|\mathcal{X}_j). \tag{14}$$

Obviously, the simplified decision-making based on the partition Φ is connected with some information loss given by the difference

$$\triangle(\Phi, \mathcal{X}) = I(\mathcal{X}, \mathcal{M}) - I(\Phi, \mathcal{M}) = H(\mathcal{M}|\Phi) - H(\mathcal{M}|\mathcal{X}) = \tag{15}$$

$$= \sum_{j \in \mathcal{M}} P(\mathcal{X}_j) H_{\mathcal{X}_j}(\mathcal{M}) - \sum_{j \in \mathcal{M}} \sum_{x \in \mathcal{X}_j} P(\boldsymbol{x}) H_x(\mathcal{M}) =$$

$$= \sum_{j \in \mathcal{M}} \sum_{x \in \mathcal{X}_j} \frac{P(\boldsymbol{x})}{P(\mathcal{X})} \sum_{m \in \mathcal{M}} q(m|\boldsymbol{x}) \log \frac{q(m|\boldsymbol{x})}{q(m|\mathcal{X}_j)} \geq 0$$

It can be seen that the last sum in the above expression represents the Kullback-Leibler information divergence which is non-negative for any two distributions $q(m|\boldsymbol{x}), q(m|\mathcal{X}_j)$ and therefore the difference $\triangle(\Phi, \mathcal{X})$ is non-negative.

In view of the above equations any cluster analysis resulting in a partition of the data space $\mathcal{X}$ is connected with some information loss from the point of view of the underlying "elementary" decision problem. In order to minimize the information loss the subsets $\mathcal{X}_j \in \Phi$ should contain the points $\boldsymbol{x} \in \mathcal{X}_j$ with similar conditional probabilities $q(m|\boldsymbol{x})$ or, in other words, having similar "elementary" properties. In this sense the the information loss (15) is a suitable criterion to control the proces of cluster analysis.

5   Minimum Information Loss Cluster Analysis

Considering the problem of cluster analysis in the framework of data-mining we are given typically a large set $\mathcal{S}$ of multivariate categorical data. In analogy with the partition Φ we define a partition $\Re$ of the set $\mathcal{S}$ by means of the decision function $d(\boldsymbol{x})$ (cf. (7))

$$\Re = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_M\}, \quad \mathcal{S}_m = \{\boldsymbol{x} \in \mathcal{S} : d(\boldsymbol{x}) = m\}, \quad m \in \mathcal{M}. \tag{16}$$

Obviously the subsets $\mathcal{S}_j \in \Re$ analogous to $\mathcal{X}_j \in \Phi$ correspond to the "elementary properties" defined by the latent class model. In view of the Bayes decision function $d(\boldsymbol{x})$, the subset $\mathcal{S}_m \in \Re$ contains only data vectors $\boldsymbol{x} \in \mathcal{S}$ with the most strongly "expressed" elementary property "m".

>From the theoretical point of view the data vectors $\boldsymbol{x} \in \mathcal{S}$ represent independent and identically distributed observations of a random vector and therefore the above information formulas are not directly applicable to the partition $\Re$. Considering the data set $\mathcal{S}$ we can estimate the statistical decision information $I(\mathcal{X}, \mathcal{M})$ contained in the latent classes by means of Eqs.

$$\hat{H}(\mathcal{M}|\mathcal{X}) = E_{\boldsymbol{x}}\{H_{\boldsymbol{x}}(\mathcal{M})\} = \frac{1}{|\mathcal{S}|} \sum_{x \in \mathcal{S}} H_{\boldsymbol{x}}(\mathcal{M}), \tag{17}$$

$$\hat{I}(\mathcal{X}, \mathcal{M}) = H(\mathcal{M}) - \hat{H}(\mathcal{M}|\mathcal{X}) = H(\mathcal{M}) - \frac{1}{|\mathcal{S}|} \sum_{x \in \mathcal{S}} H_{\boldsymbol{x}}(\mathcal{M}). \tag{18}$$

In order to evaluate the information loss connected with the partition Φ we have to estimate the information $I(\Phi, \mathcal{M})$ by means of the observation sample $\mathcal{S}$. By using the estimates

$$\hat{q}(m|\mathcal{X}_j) = \frac{1}{|\mathcal{S}_j|} \sum_{x \in \mathcal{S}_j} q(m|\boldsymbol{x}), \quad \hat{P}(\mathcal{X}_j) = \frac{|\mathcal{S}_j|}{|\mathcal{S}|}, \quad j \in \mathcal{M} \tag{19}$$

we can write

$$\hat{I}(\Phi, \mathcal{M}) = H(\mathcal{M}) - \hat{H}(\mathcal{M}|\Phi) = H(\mathcal{M}) - \sum_{j \in \mathcal{M}} \frac{|\mathcal{S}_j|}{|\mathcal{S}|} \hat{H}_{\mathcal{X}_j}(\mathcal{M}), \tag{20}$$

$$\hat{H}_{\mathcal{X}_j}(\mathcal{M}) = \sum_{m \in \mathcal{M}} -\hat{q}(m|\mathcal{X}_j) \log \hat{q}(m|\mathcal{X}_j) \tag{21}$$

and finally we obtain (cf. (15))

$$\hat{\triangle}(\Phi, \mathcal{X}) = \sum_{j \in \mathcal{M}} \sum_{x \in \mathcal{S}_j} \frac{1}{|\mathcal{S}|} \sum_{m \in \mathcal{M}} q(m|\boldsymbol{x}) \log \frac{q(m|\boldsymbol{x})}{\hat{q}(m|\mathcal{X}_j)} \geq 0 \tag{22}$$

We can estimate the information loss which will arise if we unify two distinct subsets $\mathcal{X}_j, \mathcal{X}_k \in \Phi$, i.e. the new partition Φ' will contain only one subset $\mathcal{X}_j \cup \mathcal{X}_k$ instead of the two original. Since the two partitions differ only in the modified sets, we can write in analogy with (15):

$$\triangle(\Phi, \Phi') = I(\Phi, \mathcal{M}) - I(\Phi', \mathcal{M}) = H(\mathcal{M}|\Phi') - H(\mathcal{M}|\Phi) = \tag{23}$$

Mixture	latent CA	50 clusters	20 clusters	10 clusters
Original mixture	0.00	0.00	0.00	0.00
M=10	0.06	-	-	0.06
M=20	0.01	-	0.01	0.21
M=50	0.05	0.02	0.02	0.12
M=100	0.21	0.04	0.18	0.41
M=200	0.23	0.12	0.33	0.54

Table 1: Mean value of cluster's entropies for different distribution mixtures.

$$= P(\mathcal{X}_j \cup \mathcal{X}_k)H_{\mathcal{X}_j \cup \mathcal{X}_k}(\mathcal{M}) - P(\mathcal{X}_j)H_{\mathcal{X}_j}(\mathcal{M}) + P(\mathcal{X}_k)H_{\mathcal{X}_k}(\mathcal{M}).$$

The estimate of the quantity $\triangle(\Phi, \Phi')$ based on the partition $\Re$ will depend only on the two sets $\mathcal{S}_j, \mathcal{S}_k \in \Re$. We denote (cf. (19),(21))

$$Q(\mathcal{S}_j, \mathcal{S}_k) = \frac{|\mathcal{S}_j \cup \mathcal{S}_k|}{|\mathcal{S}|}\hat{H}_{\mathcal{X}_j \cup \mathcal{X}_k}(\mathcal{M}) - \frac{|\mathcal{S}_j|}{|\mathcal{S}|}\hat{H}_{\mathcal{X}_j}(\mathcal{M}) - \frac{|\mathcal{S}_k|}{|\mathcal{S}|}\hat{H}_{\mathcal{X}_k}(\mathcal{M}). \tag{24}$$

The information loss $Q(\mathcal{S}_j, \mathcal{S}_k)$ caused by union of the sets $\mathcal{S}_j, \mathcal{S}_k \in \Re$ can be used as a criterion for the optimal choice of the pair of subsets to be unified. In other words, in each step of the procedure we unify the two sets $\mathcal{S}_j, \mathcal{S}_k \in \Re$ for which the resulting information loss $Q(\mathcal{S}_j, \mathcal{S}_k)$ is minimized.

6 Numerical Examples

For basic numerical experiments a set of artificial data was created to find out the possibilities and restrictions for the presented clustering method.

Another experiments were made with the database of handwritten numerals of Concordia University from Canada, which allows perfect visual validation of gained results.

6.1 Artificial data

In previous text we suppose that the data set $\mathcal{S}$ is a set of independent and identically distributed observations of a random vector, whose statistical properties can be described by a discrete distribution mixture (4). In order to fulfil this assumption such a mixture has been prepared and then used as a propability distribution for generating the vector data set.

The generating distribution mixture was created randomly with 10 clusters each represented by 5 similar components, vectors were binary and the dimension was 256. The size of the generated data set was 10^4 vectors.

With this data all steps of clustering algorithm were done. The table 1 shows clustering results for different parameters. For partition $\Re$ of the data set $\mathcal{S}$ (see 16) we can define a criterion of its quality like a mean value of entropies of the original classes distribution in particular cluster. In other words for every cluster we can estimate the discrete density of original classes in this cluster and calculate the entropy of this density.

When there are vectors only of one original class in a cluster the used criterion is zero and the worse the clusters are (i.e. vectors in a cluster are of various original classes) the

higher the criterion is. According to this we can see that the results in table 1 are not really good.

6.2   Numerals

Almost same experiments as in previous section were made with the handwritten numeral database of Concordia University, Canada. This database contains 6000 images of handwritten numerals from US ZIP postal codes and is usually used for supervised pattern or image recognition.

In this paper we have slightly different point of view. The images were transformed to a binary raster normalized to a defined size and we consider these images to be common binary vectors without any additional structure - that means we are interested only in the pixel values but we do not consider their positions.

This goal is quite uneasy but we are not looking for the best clustering method for this type of data; we would like to test our method in its general form. A big convenience of this data set is the possibility of visual validation of the clustering.

Figure 1: An example of the numerals data

In these experiments various vector dimensions and different number of latent classes were used. The figure 1 shows a sample from a data set with dimension 256 and the figure 2 shows the estimated distribution mixture with 100 components.
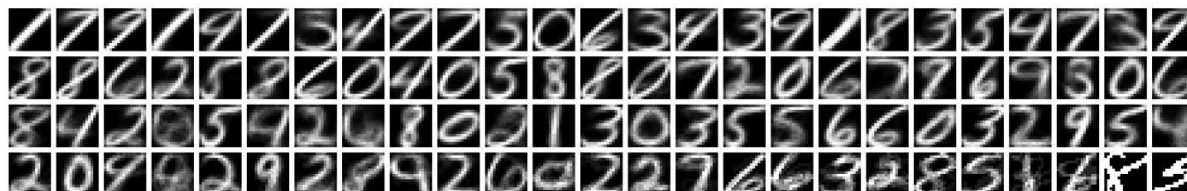

Figure 2: The estimated distribution mixture with 100 components

In the figure 3 can be seen the evolution of cluster shapes during the clustering. The first row shows 25 clusters, i.e. the state after 75th iteration. Every next row shows the state after another 5 iterations. In the most fortunate case ten different digits would be seen in the last row but in our case only few digits are recognizable.
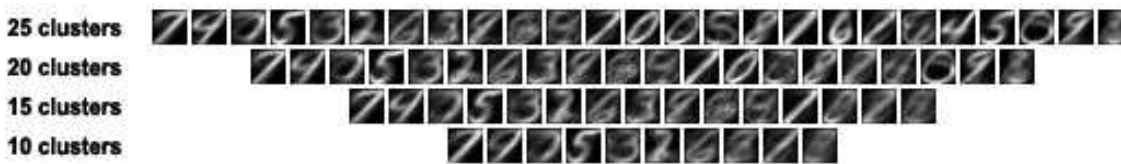

Figure 3: Some of the final steps of the hierarchical clustering

7   Conclusion

The latent class models have been used repeatedly as a tool of cluster analysis of multivariate categorical data since the standard approaches are usually not directly applicable. Unfortunately, the underlying discrete distribution mixtures with product components are not uniquely identifiable. In order to avoid the problem of identifiability the latent class model is interpreted in this paper only as a set of elementary properties. We define an information preserving transform of the original categorical data into a multidimensional real space by computing the conditional weights of the elementary properties for the data vectors $x \in \mathcal{S}$. The relations between the transformed data vectors described in terms of elementary properties are less influenced by the particular form of latent classes. The hierarchical cluster analysis of the transformed space is controlled by minimum information loss criterion.

The presented numerical experiments are not convincing but the results are not considered to be final. We are now trying to find out the problematical points of the algorithm and improve the presented method.

References

[1] Ajvazjan, S.A., Bezhaeva, Z.I., Staroverov, O.V. : Classification of Multivariate Observations, (in Russian). Moscow: Statistika (1974)

[2] Bartholomew D.J.: Factor analysis for categorical data. *J. Roy. Statist. Soc. B*, 3 **42** (1980) 293-321

[3] Blischke W.R. : Estimating the parameters of mixtures of binomial distributions. *Journal Amer. Statist. Assoc.*, **59** (1964) 510-528

[4] Boyles R.A.: On the convergence of the EM algorithm. *J. Roy. Statist. Soc., B*, **45** (1983) 47-50

[5] Carreira-Perpignan M.A., Renals S.: Practical identifiability of finite mixtures of multivariate Bernoulli distributions. *Neural Computation*, **12** (2000) 141-152

[6] Dempster, A.P., Laird, N.M. and Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. B*, **39** (1977) l-38

[7] Fielding A.: Latent structure models. In: *The Analysis of survey data*, (Eds. O'Muircheaxtaigh C.A, Payne C.), London: Wiley, (1977) 125-157

[8] Gibson W.A.: Three multivariate models: Factor analysis, latent structure analysis and latent profile analysis. *Psychometrika*, **24** (1969) 229-252

[9] Goodman L.A. : Analyzing qualitative/categorical data log-linear models and latent structure analysis. Reading, Mass.: Addison-Wesley, (1978)

[10] Grim, J. : On numerical evaluation of maximum - likelihood estimates for finite mixtures of distributions. Kybernetika, 3 **l8** (1982) 173-190

[11] Grim J.: Multivariate statistical pattern recognition with nonreduced dimensionality, *Kybernetika*, **22** (1986) 142-157

[12] Grim, J.: Design of multilayer neural networks by information preserving transforms. In: E. Pessa, M.P. Penna, A. ontesanto (Eds.), *Proceedings of the Third European Congress on System Science*, Roma: Edizzioni Kappa, (1996) 977-982

[13] Grim J.: Discretization of probabilistic neural networks with bounded information loss. In: Preprints of the 3rd European IEEE Workshop on Computer-Intensive Methods in Control and Data Processing. (Eds. Warwick K. et al.). UTIA, Czech Academy of Sciences, Prague (1998) 205-210

[14] Grim J., Boček P., Pudil P. (2001): Safe dissemination of census results by means of interactive probabilistic models. In: *Proceedings of the ETK-NTTS 2001 Conference*, (Hersonissos (Crete), European Communities 2001, **2** (2001) 849-856

[15] Grim J., Haindl M.: Texture Modelling by Discrete Distribution Mixtures. Computational Statistics and Data Analysis, 3-4 **41** (2003) 603-615

[16] Grim J., Kittler J., Pudil P., Somol P.: Multiple classifier fusion in probabilistic neural networks. *Pattern Analysis & Applications*, 7 **5** (2002) 221-233.

[17] Grim J.: Latent Structure Analysis for Categorical Data. *Research Report UTIA*, No. 2019, Academy of Sciences of the Czech Republic, Prague 2001, 13 pp.

[18] Grim J.: EM Cluster Analysis for Categorical Data. In: *Proceedings of the SSPR'06*, Hong-Kong (2006).

[19] Gyllenberg M., Koski T., Reilink E., Verlaan M.: Non-uniqueness in probabilistic numerical identification of bacteria. *Journal of Applied Prob.*, **31** (1994) 542-548

[20] Isaenko, O.K., Urbakh, K.I.: Decomposition of probability distribution mixtures into their components (in Russian). In: Theory of probability, mathematical statistics and theoretical cybernetics, Moscow: VINITI, **13** (1976)

[21] McLachlan G.J. and Peel D.: *Finite Mixture Models*, John Wiley & Sons, New York, Toronto, (2000)

[22] Lazarsfeld P.F., Henry N.W.: *Latent structure analysis*. Houghton Miflin, Boston (1968)

[23] Pearl J.: *Probabilistic reasoning in intelligence systems: networks of plausible inference*. Morgan-Kaufman, San Mateo, CA (1988)

[24] Schlesinger, M.I.: Relation between learning and self learning in pattern recognition (in Russian), Kibernetika, (Kiev), 2 (1968) 81-88

[25] Teicher, H. : Identifiability of mixtures of product measures. *Ann. Math. Statist.*, **39** (1968) 1300-1302

[26] Titterington, D.M., Smith, A.F.M. and Makov, U.E.: Statistical analysis of finite mixture distributions, John Wiley & Sons: Chichester, New York (1985)

[27] Vajda I., Grim J.: About the maximum information and maximum likelihood principles in neural networks. *Kybernetika*, **34** 4 (1998) 485-494

[28] Vermunt J.K., Magidson J.: Latent Class Cluster Analysis. In: *Advances in Latent Class Analysis*, (Eds. Hagenaars J.A., McCutcheon A.L., Cambridge University Press (2002)

[29] Xu L. and Jordan M.I. : On convergence properties of the EM algorithm for Gaussian mixtures. *Neural Computation*, **8** (1996) 129-151

Recognition of Partially Occluded Objects After Affine Transformation*

Ondřej Horáček

3rd year of PGS, email: `horacek@utia.cas.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Jan Flusser, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic

Abstract. A method dealing with recognition of partially occluded and affine transformed binary objects is presented. The method is designed for objects with smooth curved boundary. It divides an object into affine-invariant parts and uses modified radial vector for the description of parts. Object recognition is performed via string matching in the space of radial vectors.

Abstrakt. V článku je představena metoda na rozpoznávání částečně zakrytých objektů po afinní transformaci. Metoda byla vyvinuta pro objekty s hladkou hranicí. Objekt je rozdělen na afinně invariantní části a ty jsou popsány pomocí modifikovaného radiálního vektoru. K rozpoznávání je použit string matching v prostoru radiálních vektorů.

1 Introduction

Recognition of objects under partial occlusions and deformations caused by imaging geometry is one of the most difficult problems in computer vision. It is required always when analyzing 2-D images of a 3-D scene. Although many methods trying to solve this task have been published, it still remains open. Clearly, there is no universal algorithm which would be "optimal" in all cases. Different methods should be designed for different class of objects and for different groups of assumed deformations.

In this paper, we deal with objects of a curved boundary, which does not need to be polygonal – typically complicated smooth curve boundary. Further more, we assume the objects are deformed by an unknown affine deformation. This assumption is realistic in most real situations when photographing an object arbitrary oriented in the 3-D space. Precisely, such a deformation is a perspective projection but it is well known it can be approximated by affine transform when the object-to-camera distance is large comparing to the size of the object.

We introduce a method developed for the recognition under the mentioned conditions. First, the shape is divided into parts which are defined by means of inflection points of the object boundary. Then the shape of each part is described by a special kind of radial vector. Finally, the parameters of the affine deformation are estimated and classification is performed by string matching in the space of radial vectors. The performance of the method is demonstrated by experiments.

---

2 Overview of current methods

Current methods can be classified into two major categories. The methods of the first group divide the object into affine-invariant parts. Each part is described by some kind of "standard" global invariants, and the whole object is then characterized by a string of vectors of invariants. Recognition under occlusion is performed by maximum substring matching. Since inflection points of the boundary are invariant to affine (and even projective) deformation of a shape, they become a popular tool for the definition of the affine-invariant parts. This approach was used by Ibrahim and Cohen [3], who described the object by area ratios of two neighboring parts. As a modification which does not use inflection points, concave residua of convex hull could be used (Lamdan [5]). For polygon-like shapes, however, inflection points cannot be used. Instead, one can construct "cuts" defined by three or four neighboring vertices. Yang and Cohen [12] used area ratios of the cuts to construct affine invariants. Flusser [2] further developed their approach by finding more powerful invariant description of the cuts. A similar method was successfully tested for perspective projection by Rothwell [7].

The methods of the second group are "intrinsically local", i.e. they do not divide the shape into subparts but rather describe the boundary in every point by means of its small neighborhood. In that way they transform the boundary to a so-called signature curve which is invariant to affine/projective transform. Recognition under occlusion is again performed by substring matching in the space of signatures. Typical representatives of this group are differential invariants. They were discovered hundred years ago by Wilczynski [11] who proposed invariant signatures based on derivatives of up to 8-th order. Weiss [10] introduced differential invariants to the computer vision community. He published a series of papers on various invariants of orders from four to six [9, 1]. Although differential invariants seemed to be promising from theoretical point of view, they have been experimentally proven to be extremely sensitive to inaccurate segmentation of the boundary, discretization errors and noise.

Following methods dealing with recognition of transformed object could be relevant to our conditions, too. Mokhtarian and Abbasi [6] used inflection points themselves to characterize the boundary. They constructed so-called Curvature Scale Space and traced the position of inflection points on different levels of image pyramid. The trajectories of the inflection points then served as object descriptors. Lamdan [5] used mutual position of four "interesting" points for the recognition. To verify the received match, normalized concave areas were described by the radial vector. There are also methods based on wavelet transform of the boundary. Tieng at al. [8] introduced wavelet-based boundary representation, where affine invariance was achieved by enclosed area contour parameterization. A similar approach was used by Khalil at al. [4]. However, the use of the wavelet-based methods in case of partial occlusions is questionable.

3 Definition of affine-invariant parts

Both inflection points and central points of straight lines are affine invariant, i.e. the properties "to be an inflection point" and "to be a central point of a straight line" are preserved under arbitrary nonsingular affine transform. Thus, we use these points (called

"cut points" in the sequel) for the construction of affine-invariant parts. We connect each pair of neighboring cut points by a line. This line and the corresponding part of the object boundary form a convex region which may but need not to lie inside the original object (in Fig. 1c). If further, we call these regions the cuts and the parts equivalently. The sequence of such cuts carries efficient information about the object.

Detection of inflection points of discrete curves has been discussed in numerous papers. Let us recall that in the continuous domain an inflection point is defined by a constraint $\ddot{x}(t)\dot{y}(t) - \dot{x}(t)\ddot{y}(t) = 0$, where $x(t), y(t)$ represent a parameterization of the curve and the dots denote derivatives with respect to t. When this definition is directly converted to the discrete domain, it becomes very sensitive to sampling and noise. Thus, we propose a new robust method of curvature estimation.

A circle with a fixed radius is placed on each boundary point (in Fig. 1a). Ratio of the whole circle area to its area contained in the object serves for estimation of the curvature. When this ratio equals 0.5, the boundary has zero curvature and the corresponding point is either an inflection point or it lies inside a straight segment. From the obtained values, we construct a curvature graph (in Fig. 1b) and smooth it. Now we find the division of the original object into parts: We define cut points as zero-crossing points and middle points of approximately zero-value segments. Furthermore, a request of sufficient cut size is considered: the segment of the curvature graph between two cut points should have sum of values above some threshold, otherwise it is treated as a part of zero curvature.
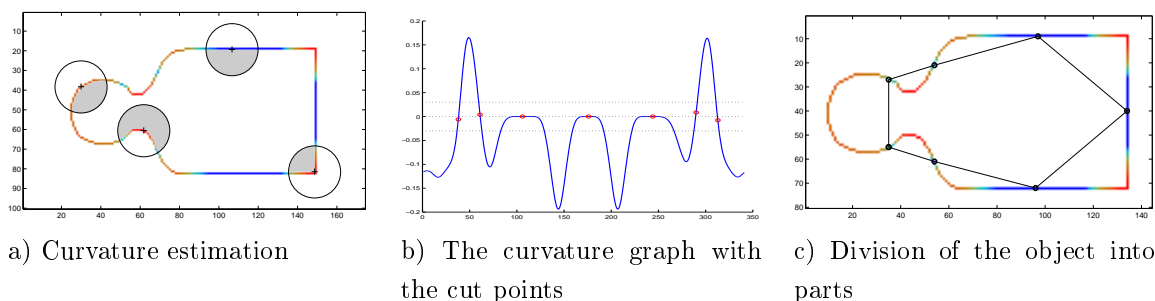

a) Curvature estimation | b) The curvature graph with the cut points | c) Division of the object into parts

Figure 1: Definition of affine-invariant parts

4   Description of the parts

The object is represented by parts called cuts as described in the previous section. By adding a description of the shape of the individual cuts we get a description of the whole object which is robust to occlusion. Robustness to occlusion means that if some part of the object boundary is missing or changed, only few elements of the feature vector are changed. This is an important attribute. Note that traditional global methods, for instance description of the object by moment invariants or Fourier descriptors, do not have this property.

It would be possible to describe each cut individually and eliminate the impact of the deformation by using proper affine invariants (moment invariants or Fourier descriptors for instance). In such case, however, we do not employ important information that all the

cuts were deformed by *the same* transformation. Including this consistency information in the object description can significantly increase the recognition performance. Thus, we propose the following description of the cuts by a modified radial vector, with included position of control points. See complete demo object description in Fig. 2a.

The spokes of the modified radial vector come from the middle of the cutting line and they divide the cut into subparts of equal area. For each cut, they are constructed as follows.

1. Define the desired number n of the spokes (i.e. the length of the radial vector).

2. Go through the outer boundary of the cut.

3. For each step calculate the area of the triangle between the neighbouring boundary points and the midpoint of the cutting line.

4. If the cumulated area exceeds $k/(n-1)$ fraction of the total cut area, the k-th spoke ends in the current boundary point.

The introduced modified radial vector divides the cut invariantly under affine transformation. Note that a classical radial vector with constant-angle spokes distribution or constant-boundary length spokes distribution has not such a favorable property.
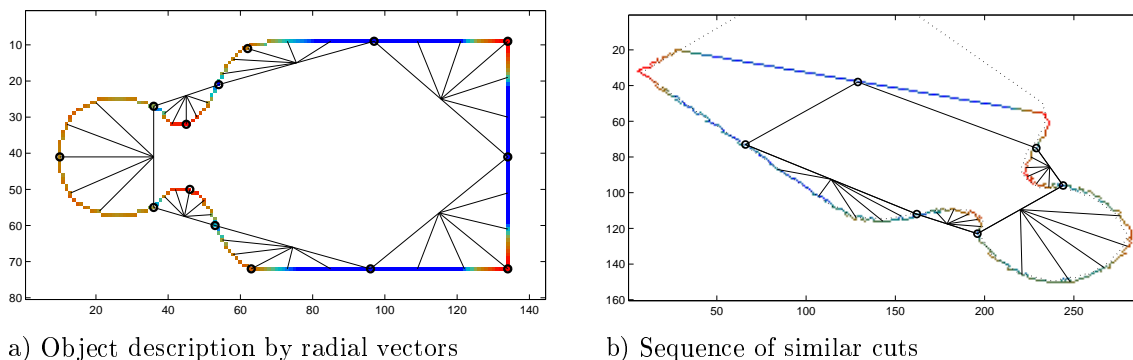

a) Object description by radial vectors   b) Sequence of similar cuts

Figure 2: Description and matching of the demo object

5   Matching

The image is classified by finding the longest and best matching section of the border (in Fig. 2b). This is realized by comparing sequences of parts between the classified image and database objects.

1. For each cut of the database object and each cut of the image do the following:

2. Take a sequence of cuts starting from the current cut. A length of this sequence is gradually incremented, it begins with the length of the previous longest successful match (or equals one in case of the first trial).

3. Recover the affine transformation T between the database and classified object. Use the least square fit of the control points of all cuts involved in the sequence. Each cut has three control points: two cut points and the middle-spoke end-point.

4. Transform the database radial vectors by the transformation T.

5. Compare the transformed database radial vector sequence with the current one from the image. Similarity measure S is evaluated for this purpose – see below. If S is smaller than required similarity threshold, this sequences are considered not to match, and a start of next sequence is taken on the step 1.

6. The current sequence of radial vectors was successfully tested for a match in the previous step. If this sequence is longer than the previous best match, or is equal and the similarity level S is higher, select this match as the best one. Now try to make the matching sequence even longer – continue with the step 2.

Similarity measure S is introduced for suitable comparison of the radial vectors u, v. $S = 1$ only if $u = v$, S approaches zero for growing vector difference. The single similarity measure s_i of the i-th spoke lengths u_i, v_i is a Gaussian quantity of the $u_i - v_i$ difference (in Fig. 3a)

$$s_i = e^{-\frac{1}{\sigma_i^2}\left(\frac{u_i - v_i}{2}\right)^2}, \qquad \sigma_i = k_1 + k_2 \left|\frac{u_i + v_i}{2}\right|.$$

The Gaussian dispersion σ_i has an absolute component k_1 which realizes a noise tolerance and relative one, constant k_2, which determines a tolerance relative to the value size.

We have the following requirements for combining single component s_i to overall similarity measure S: $S = s_i$ if all s_i are equal, $S = 0$ if at least one $s_i = 0$, and S needs to be sensitive to all s_i. Moreover, it is reasonable to require S to be 0.75 if all but one s_i equal 1 and one s_i equals 0.5 (in Fig. 3b). All these criteria are met for example by weighted average with weights w_i inversely proportional to s_i

$$S = \frac{\sum_{i=1}^n w_i \cdot s_i}{\sum_{i=1}^n w_i}, \qquad w_i = \frac{n-2}{s_i} - (n-3).$$

6 Experimental results

The proposed method was tested on a set of 24 binary objects (Fig. 4) segmented from original color images. The objects were successively deformed by various affine transforms, their various regions were occluded and then the objects were matched against the original database. As a matching criterion which should be maximized we took the number of those cuts of the test object which match with the cuts of the database object. This is in fact a well-known principle of string matching.

For illustration, two examples are shown in Fig. 5. On the left-hand side, one can see partially occluded and transformed objects. The corresponding database objects (which were successfully found in both cases) are shown on the right-hand side. The control (inflection) points are highlighted, their connecting lines define the division into cuts.
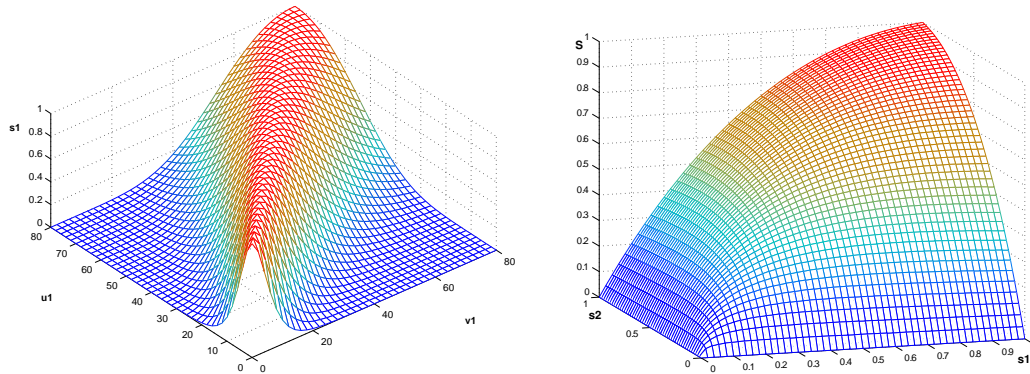
a) Single spokes similarity measure s_i depends on difference of the spoke lengths

b) Total similarity measure of radial vectors S is a combination of the spoke similarity measures

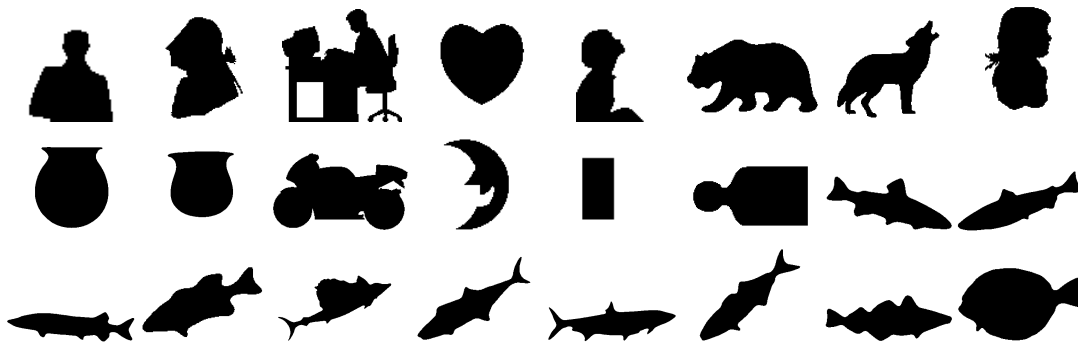Figure 3: Similarity measure is introduced for radial vectors comparison

Figure 4: Our object database

The spokes of the corresponding radial vectors are drawn inside the matched cuts of the image.

The modified radial vector describes the boundary with a good precision, the tolerance to a shape perturbations is controlled by user-defined parameters/thresholds. This enables an optimization for various types of shapes. Surprisingly, the boundary does not need to be a smooth curve with well-defined inflection points. The method finds control points even on polygonal cuts (in Fig. 5a). Furthermore, due to some tolerance threshold for the detection of inflection points, we can obtain even some non-convex cuts. We are able to construct radial vector also for these non-convex cuts (in Fig. 5d). Remind that our modified radial vector is created by dividing cumulative area while proceeding throw the cut boundary.

The object description and the result of a recognition naturally depends on the conditions of the experiment: character of the shapes, amount of occlusion, degree of the transformation, and noise. Before summing up statistical experiments, let's focus on some situations in detail. As we can see in Fig. 6b), thanks to our robust curvature esti-

a) Deformed object, overlapped by a square. Radial vector is drawn in matching cuts

b) Recognized database object

c) Partially occluded and deformed image

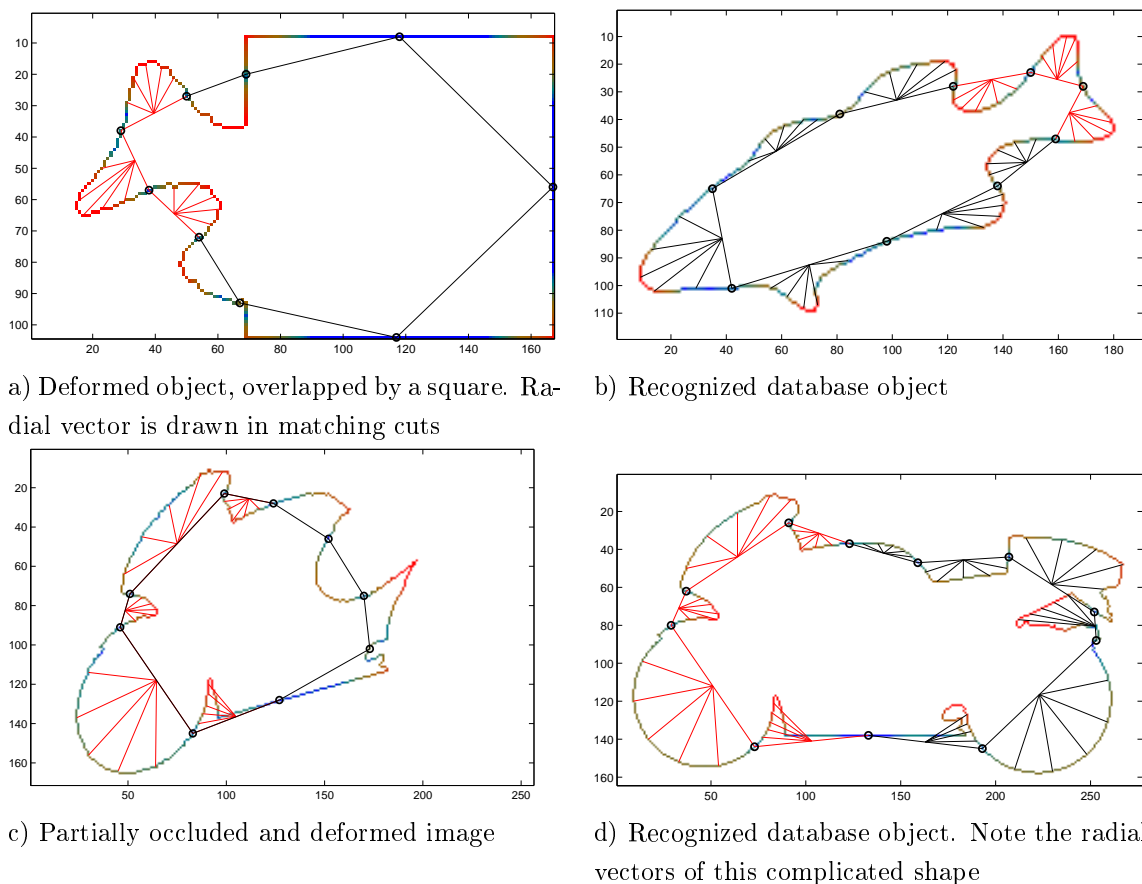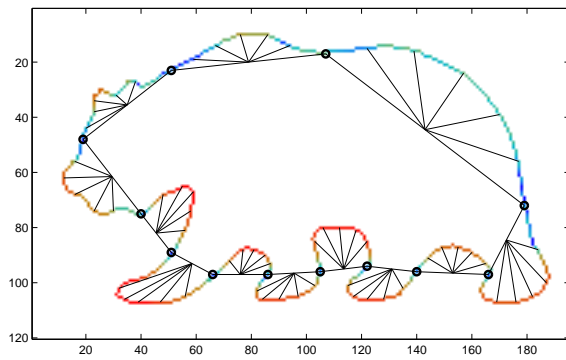d) Recognized database object. Note the radial vectors of this complicated shape

Figure 5: Recognition examples and description of recognized database objects

mation and similarity measure, the proposed method's resistance to noise is quite good. Affine transformation was applied to the images in Fig. 6c) and 6d). The original object was recognized from the image, but some cuts were not included in the match. We will explain this phenomenon in the following example.

At the bottom of Fig. 7 are two overlapped objects, the second one was transformed by a slightly wilder transformation. One can see that the position of the marked control point was evaluated differently due to changed curvature of the boundary. Therefore, one of the cuts changed its shape and did not match with the pattern cut. This leads to match reduction and worse position detection (the overlayed database object is drawn by dotted line). Control point instability is caused by unsuitable shapes (without clear inflection points), affine transformation (affects the curvature), or occlusion (inflection points originally ignored can become significant). Although our control point detection on a smooth boundary was improved comparing to traditional methods, it still remains a principal problem. In general, we can say that the recognition is as good as the stability of the critical points.

Recognition under various conditions is summarized in Table 1. "Image area" denotes the size of the visible part of the test object (in per cent), "Constant scale of details" indicates whether or not the same thresholds were used for database and test objects when detecting inflection points, and "Transformation" means the significance of the deforma-
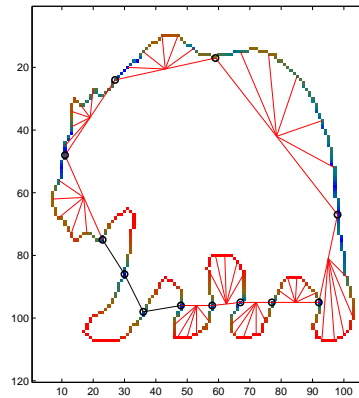
a) Description of the original object
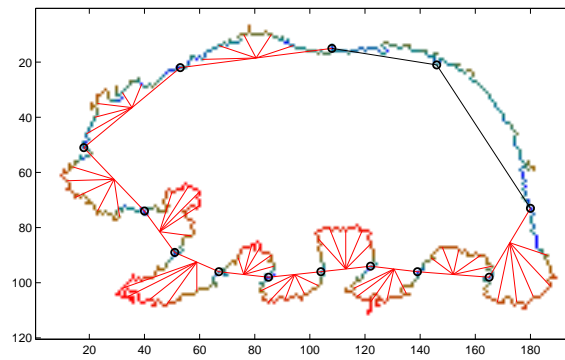
b) Image with noise was recognized well
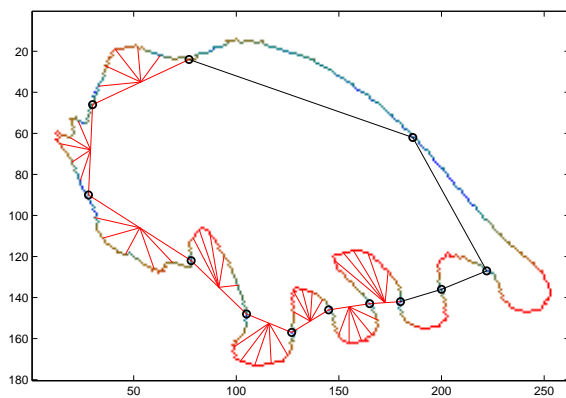
c) Transformed image was recognized, even if the back cut did not match

d) The match was shorten by image transformation

Figure 6: Influence of affine transformation and noise to recognition

Database object

Control points are stable for this occlusion and transformation

Instability of a control point leads to match reduction and worse position detection
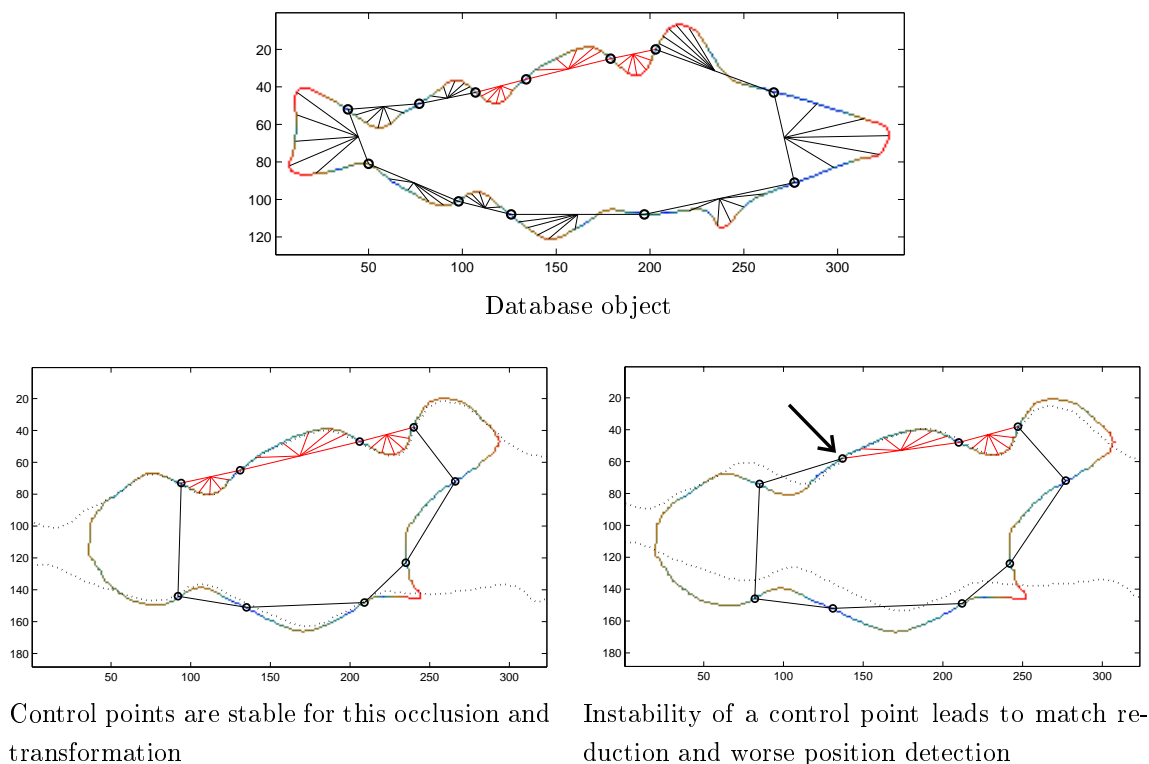
Figure 7: The impact of instability of the control points

tion measured by skewing. The table itself shows the maximum number of matching cuts over all database objects. In all instances where the maximum number of matching cuts was greater than two the test objects were recognized correctly. One or two matching cuts does not ensure unique correct match, so the classification can be wrong. These not-recognized objects are represented inside the table by image. Their problem is caused by strong deformation or large amount of occlusion which leads to instability of control points.

6.0.1 Comparison with area ratio method

The presented method is compared to Ibrahim and Cohen [3] paper, which is based on area ratio of shape parts (cuts). Although their algorithm was originally not developed for recognition of partially occluded objects, it is suitable for these conditions too. Their cuts are bounded by inflection points as well.

Recognition power and discriminability of the methods were tested by mutual matching of our 24 database objects (in Fig. 4). Implementations of both algorithms use the same detection of inflection points, therefore we can set a threshold of the Ali's method to classify our correct matches as good ones. In Table 2, counts of incorrectly matched neighbouring cuts are compared.

It is clear that the area ratio carries much less information than our modified radial vector. Proposed method needs only 3 cuts for unique correct match, while the area ratio method requires 5 cuts. These numbers are relevant to our database, different number of

Table 1: Experiment of 8 object recognition under various conditions. Correct recognition is represented by the numbers of matching cuts. Icons inside the table denote cases whet insufficient number of cuts were found unique recognition.

Image area	100%	90%	50%	50%	100%	100%	50%
Constant scale	yes	yes	no	yes	yes	yes	yes
Transformation	none	none	none	none	medium	strong	medium
Original image	(icon)	(icon)	(icon)	(icon)	(icon)	(icon)	(icon)
(bear)	12	7	4	3	10	7	3
(wolf)	11	8	(icon)	7	4	6	(icon)
(motorcycle)	11	8	3	4	6	4	3
(person)	11	9	3	4	3	(icon)	3
(bottle)	7	4	(icon)	(icon)	7	7	(icon)
(fish)	13	8	(icon)	(icon)	3	(icon)	(icon)
(fish)	10	7	(icon)	6	8	(icon)	4
(fish)	9	6	4	4	4	8	4

Table 2: Number of incorrect matches of area ratio and proposed method. Thresholds of Ali's method was set to classify our correct matches as good ones.

Length of possible match	Area ratio method	The proposed method
4-cut string	12 wrong matches	0 wrong matches
3-cut string	59 wrong matches	0 wrong matches
2-cut string	249 wrong matches	10 wrong matches

matching cuts could be required for unique object match on some other database. Both methods can be affected by control points instability. In Fig. 8 you can see one of the 10 worst two-cut wrong matches of presented method and a sample of wrong four-cut match of area ratio method.

Two-cut wrong match of the proposed method

Four-cut wrong match of the area ratio method
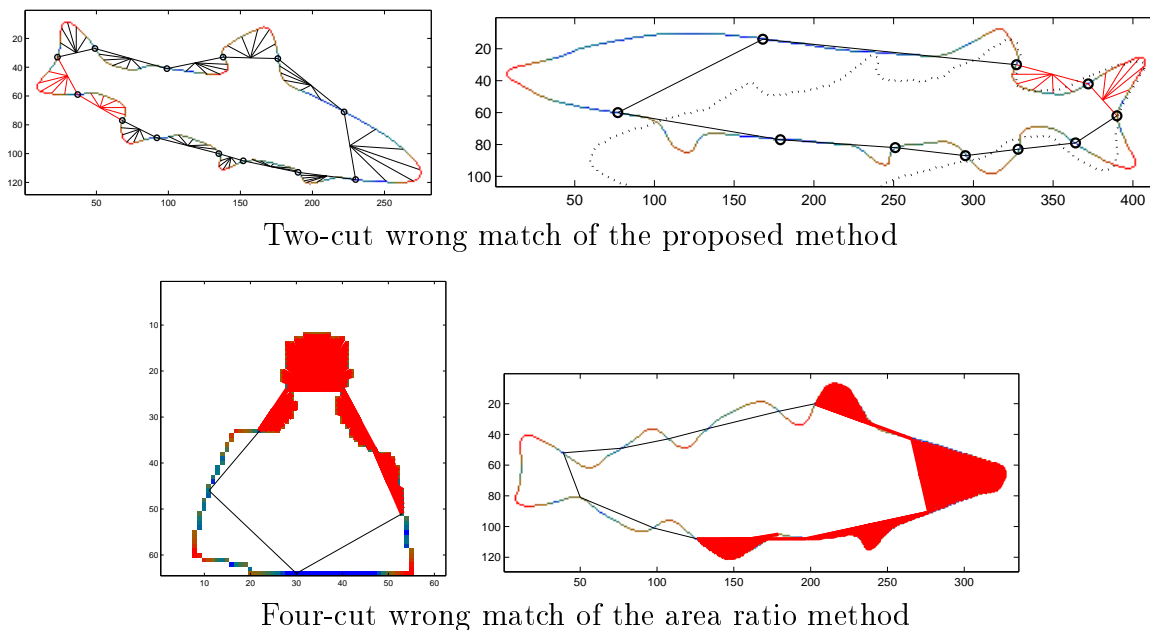
Figure 8: Example of too short matches for correct recognition, for both compared methods. Proposed method needs 3 cuts for unique correct match, the area ratio method requires 5 cuts.

7   Conclusion

We presented a method for recognition of partially occluded binary objects deformed by affine transformation. The method uses local affine-invariant description of the object boundary by means of inflection points and radial vectors. When working with digital boundary, the major limitation of the method is stability of inflection points. As the experiments demonstrated, if the curve has "prominent" inflection points, they are usually very stable under affine transformation and the method works perfectly. On the other hand, in the case of obscure boundary the inflection points may be detected at different positions depending on the particular transformation and/or occlusion and the recognition may fail.

Our experiment proved a good discrimination power of the method. On given test set, we discovered that if the maximum number of matched boundary parts between the unknown object and the database is greater than 2, it always indicated a correct match. Thus, this threshold can be recommended for prospective real experiments too.

References

[1] A. M. Bruckstein, E. Rivlin, and I. Weiss. *Scale space semi-local invariants*. Image Vision Computing **15** (May 1997), 335–344.

[2] J. Flusser. *Affine invariants of convex polygons*. IEEE Transactions on Image Processing **11** (2002), 1117–1118.

[3] W. Ibrahim and F. S. Cohen. *Registering histological 2D sections of a rat brain with a 3D brain atlas using geometric curve invariants*. IEEE Trans. Med. Imaging **17** (1998), 957–966.

[4] M. I. Khalil and M. M. Bayoumi. *A dyadic wavelet affine invariant function for 2D shape recognition*. IEEE Trans. Pattern Anal. Mach. Intell **23** (2001), 1152–1164.

[5] Y. Lamdan, J. T. Schwartz, and H. J. Wolfson. Object recognition by affine invariant matching. In 'IEEE Computer Vision and Pattern Recognition or CVPR', 335–344, (1988).

[6] F. Mokhtarian and S. Abbasi. *Shape similarity retrieval under affine transforms*. Pattern Recognition **35** (2002), 31–41.

[7] C. Rothwell, A. Zisserman, D. Forsyth, and J. Mundy. *Geometric Invariants in Computer Vision*. MIT Press, Cambridge, (1992).

[8] Q. M. Tieng and W. W. Boles. *An application of wavelet-based affine-invariant representation*. Pattern Recognition Letters **16** (1995), 1287–1296.

[9] I. Weiss. Projective invariants of shape. In 'IEEE Computer Vision and Pattern Recognition or CVPR', 291–297, (1988).

[10] I. Weiss. *Noise-resistant invariants of curves*. IEEE Trans. Pattern Anal. Mach. Intell **15** (1993), 943–948.

[11] E. Wilczynski. *Projective Differential Geometry of Curves and Ruled Surfaces*. B. G. Teubner, Leipzig, (1906).

[12] Z. Yang and F. S. Cohen. *Image registration and object recognition using affine invariants and convex hulls*. IEEE Transactions on Image Processing **8** (1999), 934–946.

Binary Object Recognition Using Polygonal Approximation[*]

Jan Kamenický

3rd year of PGS, email: `j.kamenicky@sh.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Jan Flusser, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic

Abstract. A method dealing with recognition of partially occluded and affine transformed binary objects is presented. The method is designed especially for objects with polygon-like shaped boundary. It approximates the boundary with a polygon and uses it as the object's description. Object recognition is then performed via special clustering in the space of affine transformation parameters. Results are demonstrated on synthetic as well as real experiments.

Abstrakt. V této práci je popsána metoda řešící rozpoznávání částečně zakrytých a afinně deformovaných binárních objektů. Metoda je navržena zejména pro objekty s hranicí polygonálního tvaru. Aproximuje hranici polygonem a tento používá k popisu vlastního objektu. Vlastní rozpoznávání je poté prováděno specifickým shlukováním v prostoru parametrů afinních transformací. Výsledky jsou demonstrovány na umělých i reálných experimentech.

1 Introduction

Recognition of objects under partial occlusions and deformations caused by imaging geometry is one of the most difficult problems in computer vision. It is required when analyzing digital images of a 3-D scene. Though many methods trying to solve this task have been published, it still remains open. Clearly, there is no universal algorithm which would be "optimal" in all cases. Different methods should be designed for different classes of objects and for different groups of assumed deformations. In this paper we assume the objects are deformed by an unknown affine deformation. This assumption approximates real photos with a week perspective deformation.

We introduce a method developed for the recognition of polygon-like shaped objects. First, the boundary is approximated by a polygon which is formed by longest linear segments of the boundary. When comparing two such polygons, affine transformation parameters between corresponding parts of the polygons are computed and used for classification. The performance of the method is demonstrated by experiments on synthetic as well as real data.

The method is intended for binary objects. In case of graylevel/color images, the input image is assumed to be segmented first.

2   Overview of current methods

Current methods can be classified into two major categories. The methods of the first group divide the object into affine-invariant parts. Each part is described by some kind of "standard" global invariants, and the whole object is then characterized by a string of vectors of invariants. Recognition under occlusion is performed by maximum substring matching. Since inflection points of the boundary are invariant to affine (and even projective) deformation of a shape, they become a popular tool for the definition of the affine-invariant parts. This approach was used by Ibrahim and Cohen [10], who described the object by area ratios of two neighboring parts. As a modification which does not use inflection points, concave residua of convex hull can be used. For polygon-like shapes, however, inflection points cannot be used. Instead, one can construct "cuts" defined by three or four neighboring vertices. Yang and Cohen [5] used area ratios of the cuts to construct affine invariants. Flusser [2] further developed their approach by finding more powerful invariant description of the cuts. Similar method was successfully tested for perspective projection by Rothwell et al. [4].

Lamdan [7] used mutual position of four "interesting" points for the recognition. To verify the received match, normalized concave areas were described by radial vector.

The methods of the second group are "intrinsically local", i.e. they do not divide the shape into subparts but rather describe the boundary in every point by means of its small neighborhood. In that way they transform the boundary to so-called signature curve which is invariant to affine/projective transform. Recognition under occlusion is again performed by substring matching in the space of signatures. Typical representatives of this group are differential invariants. They were probably discovered by Wilczynski [8] who proposed invariant signatures based on derivatives of up to eight order. Weiss introduced differential invariants to the computer vision community. He published a series of papers [6], [9], [11] on various invariants of orders from four to six. Although differential invariants seemed to be promising from theoretical point of view, they have been experimentally proven to be extremely sensitive to inaccurate segmentation of the boundary, discretization errors and noise.

Mokhtarian and Abbasi [12] used inflection points themselves to characterize the boundary. They constructed so-called Curvature Scale Space and traced the position of inflection points on different levels of image pyramid. The trajectories of the inflection points then served as object descriptors.

There have been also methods based on wavelet transform of the boundary. Tieng and Boles [3] introduced wavelet-based boundary representation, where affine invariance was achieved by enclosed area contour parametrization. A similar approach was used by Khalil and Bayeoumi [1]. However, the use of the wavelet-based methods in case of partial occlusions is questionable.

3   Proposed method

The method itself can be divided into two main parts. The first is polygonal approximation of the object boundary which can be used on any planar continuous and closed curve. The second is classification of objects represented by an ordered set of points (in

2-D, i.e. polygons) invariant to affine deformation and multiple partial occlusions.

3.1   Polygonal approximation

Most of existing methods of polygonal approximation are restricting vertices of the polygon only to the boundary points. In the proposed method, on the other hand, the vertices can lay outside (or inside) the boundary. This approach makes the algorithm of the approximation more complicated and slower (more computing time is required), but also more precise, and the final polygon is more stable (see Fig. 1).

A brief description of the algorithm is as follows.

1. For each boundary point, we compute longest linear segment from that point:

    (a) As approximated set of points we take the initial border point and the two of its successive neighbours in the clockwise order.

    (b) Using least squares method, we compute a line that best approximates the set of points.

    (c) We compute the error of the approximation (i.e. the average of the squared Euclidian distances of approximated points from the line), and if it is smaller than a preselected threshold, we add another neighbouring border point in the clockwise order and repeat from (b).

2. Now we form the final polygon from these approximating lines:

    (a) We begin with the longest line (the line corresponding to the largest approximated set of border points).

    (b) We select line that extends the approximated set of points most of all, but only if the resulting point of intersection of the two lines is acceptable. The acceptability is determined by the relative distances of the intersection from the projections of the marginal points to the appropriate lines. These distances $|B_1P|/|A_1B_1|$ and $|B_2P|/|B_2C_2|$ (see Fig.2a) should not exceed a preset threshold (a value of 30% can be used).

    (c) Repeat (b) until the polygon is closed.

Main problem of this algorithm is, that the final polygon is not invariant to scaling and affine deformation. We can partially overcome this problem, when we can estimate the amount of scaling, by using respectively adjusted threshold. However, it is not difficult to realize, that the invariantness and stability are going against each other. So we have to choose, whether we want a stable approximation, or a more invariant one, which will be very sensitive to noise and inaccurate segmentation.

3.2   Classification

Given the representing polygon from previous section, we can now deal with classification, using a special algorithm that takes the partial occlusion of the polygon into consideration.

Suppose we have two polygons: model and compared. The number of their vertices are n and m respectively.
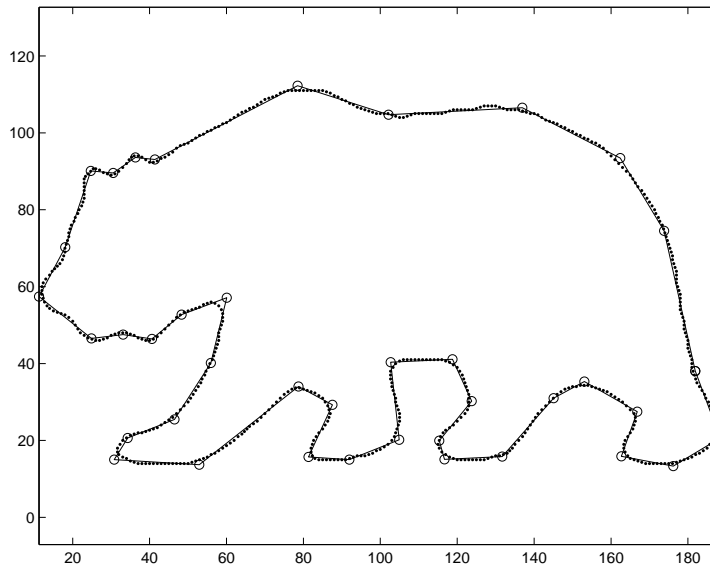
Figure 1: Approximating polygon
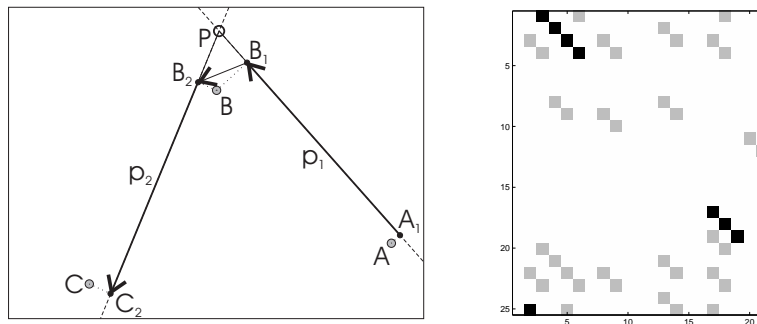
Figure 2: (a) Acceptability of the intersection   (b) Sequences

1. For each vertex of the model polygon i and each vertex of the compared polygon j, we compute six parameters of affine transformation between points $i-1, i, i+1$ and $j-1, j, j+1$. So we get $n \times m \times 6$ matrix.

2. We decompose these parameters $\mathbf{A}$ to basic transformations - skewing $\mathbf{S}_k$, scaling $\mathbf{S}_{xy}$, rotation $\mathbf{R}_\varphi$ and translation $\mathbf{T}_{xy}$.

$$\mathbf{A} = \left(\begin{array}{ccc} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ 0 & 0 & 1 \end{array} \right)$$

$$\mathbf{A} = \mathbf{T}_{xy} \cdot \mathbf{R}_\varphi \cdot \mathbf{S}_{xy} \cdot \mathbf{S}_k = \left(\begin{array}{ccc} s_x \cos\varphi & s_k s_x \cos\varphi - s_y \sin\varphi & t_x \\ s_x \sin\varphi & s_k s_x \sin\varphi + s_y \cos\varphi & t_y \\ 0 & 0 & 1 \end{array} \right)$$

3. We compare parameters at position i, j with only the diagonal neighbour at position $i+1, j+1$. We consider the parameters not similar, when the difference of

either of the parameters $(s_x, s_y, s_k, \varphi, t_x, t_y)$ is larger than a corresponding threshold (determined experimentally). We form a new binary $n \times m$ "similarity" matrix representing these similar elements.

4. Now we find sequences of similar parameters that represent corresponding parts of the polygons (see Fig.2b), compute affine transformation parameters for these sequences and link sequences with similar parameters together.

5. Finally we compute the relative part of the polygon that is matched by each found set of sequences and select the best match.

For better results, we can also include other supporting criterion for eliminating false matches. We apply inverse affine transformation to the compared polygon, and calculate the relative area, which falls inside the area of the model polygon. When this gives less than for example 90%, it is very probable, that this is a false match.

We obtain the final result of classification by comparing the compared polygon with all model polygons and determining the best match using previously defined criterions.

4 Experimental results

In Fig.3, we can see an example of matching of a partially occluded polygonal object. Here we can see several aspects of the proposed method.

(a) model   (b) compared object   (c) matching result

(d) similarity matrix   (e) similarity of sequences
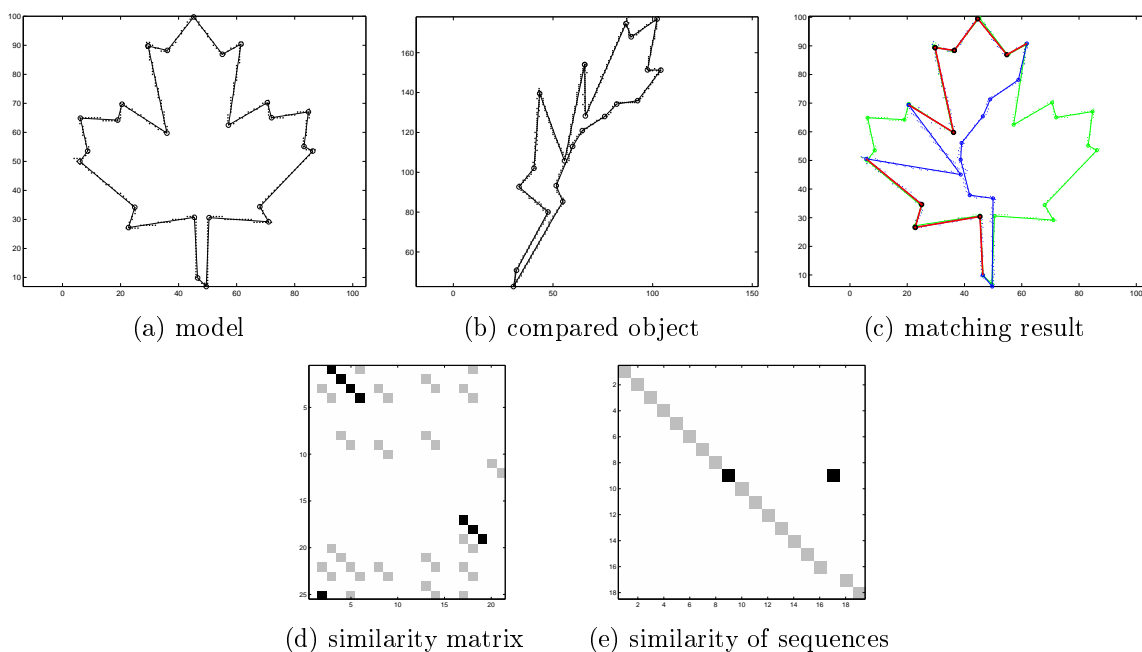
Figure 3: Matching of partially occluded polygonal object

At first, for polygonal objects, the polygonal representation is quite stable, even after affine transformation (this stability is not so good for non-polygonal objects). This stability includes also the fact, that the occlusion damages only immediately neighbouring

vertices. How we can see, actual classification gives good results, even with multiple occlusions. This is mainly because of uniting similar found sequences together (see Fig.3e).

(1)   (2)   (3)

(4)   (5)   (6)
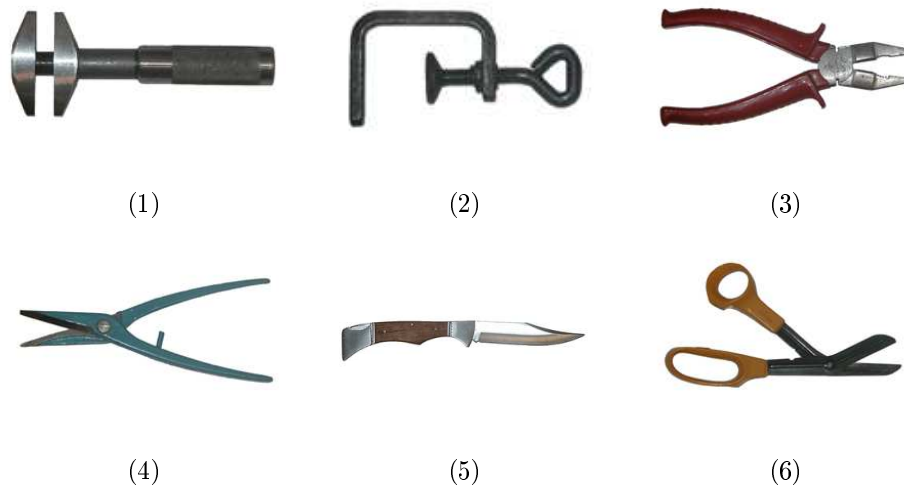
Figure 4: Model objects

(1)   (2)   (3)

(4)   (5)   (6)
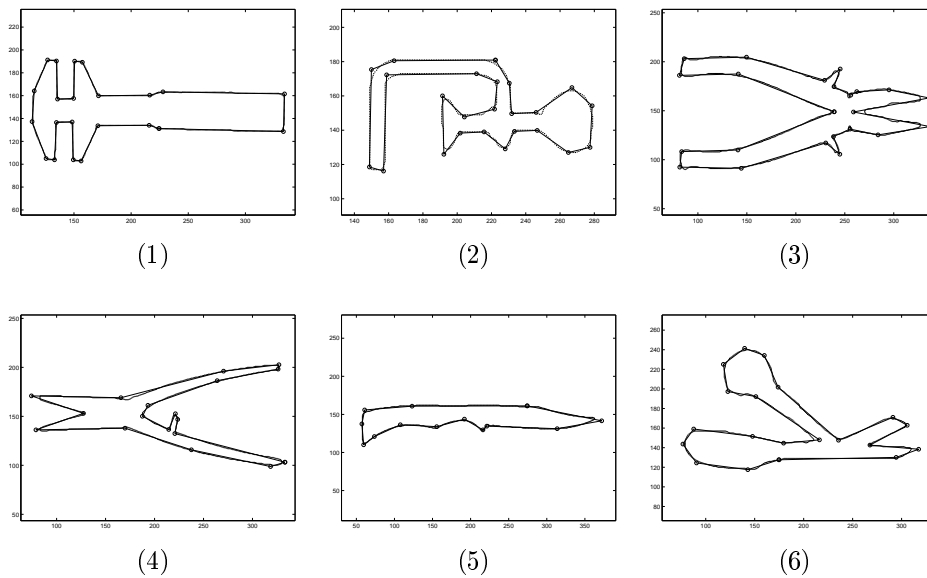
Figure 5: Binary objects and approximating polygons

Now we will present experimental results on real images. The tested set consisted of six models (see Fig.4), which were segmented to binary objects just by simple thresholding. For these binary objects, approximating polygons were computed (see Fig.5).

As an illustrative example, we will use the compared image and its polygonal representation shown in Fig.6. As we can see, the matching of model object (3) is quite stable, especially because of multiple sequences forming the match. The shape of the

border of the model (6) is quite rounded (non-polygonal) and therefore the stability of the approximating polygon is not so good. That leads to slightly worse matching of this object, but still acceptable. In case of model (1), we can see, that the objects is matched, in spite of not actually being present in the compared image. In fact this match is not a nonsense, but only corresponds to very strong affine transformation (especially scaling of factor about 7).

(a) original image   (b) binary representation

(c) model (1)   (d) model (3)   (e) model (6)
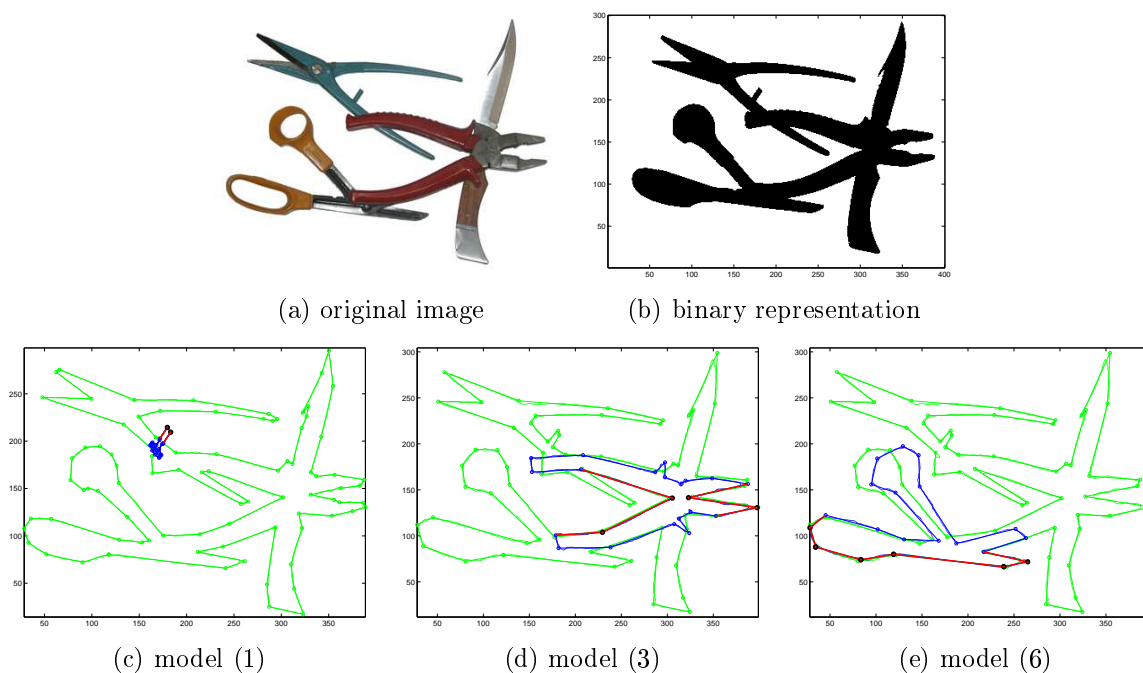
Figure 6: Compared image and examples of matching

Complete results of matching of all model objects with the compared image can be seen at Table 1. We can see, that the used matching criterion is not the most relevant feature for deciding whether the object matches. In real conditions it is better to adapt the criterion to special features present in the scene for achieving best possible results.

	sequence length	matching criterion	border part	area part	skewing	scaling x	scaling y
model (1)	2	27.8%	35.2%	94.8%	0.51	0.13	0.17
model (2)	2	10.4%	21.5%	87.0%	0.13	0.93	-0.07
model (3)	2 + 2	35.7%	43.7%	95.4%	-0.30	0.89	0.89
model (4)	12	63.4%	70.4%	97.5%	-0.02	0.95	0.95
model (5)	2	31.3%	36.6%	96.4%	0.66	0.87	0.82
model (6)	6	9.1%	36.9%	81.2%	0.26	1.00	1.04

Table 1: Matching results for compared image

5   Conclusion

We presented a method for recognition of partially occluded binary objects deformed by affine transformation. The method uses polygonal approximation of the object boundary as the object description. The major limitation of this approach is the stability of the polygon when dealing inaccurate segmentation of the object and different affine deformations (especially scaling and skewing). As the experiments demonstrated, if the curve has polygon-like shape, the approximation is usually stable and the method works well. On the other hand, in case of smooth boundary interrupted by partial occlusion the approximation may result in a much different polygon and the recognition may fail.

Given good input data (polygons), the classification itself is very stable and gives good results even under heavy conditions (affine deformations and multiple occlusions).

References

[1] M.I. Khalil, M.M. Bayeoumi. *A Dyadic Wavelet Affine Invariant Function for 2D Shape Recognition.* IEEE Transaction on PAMI, vol. 23, no. 10 (2001), 1152–1163.

[2] J. Flusser. *Affine Invariants of Convex Polygons.* IEEE Transactions on Image Processing, vol. 11, no. 9 (September 2002).

[3] Q.M. Tieng, W.W. Boles. *An Application of Wavelet-based Affine-invariant Representation.* Patern Recognition Letters, vol. 16 (1995), 1287–1296.

[4] C.A. Rothwell, A. Zisserman, D.A. Forsyth, J.L. Mundy. *Fast Recognition Using Algebraic Invariants.* Geometric Invariants in Computer Vision, Cambridge, MIT Press (1992), 398–407.

[5] Z. Yang, F.S. Cohen. *Image Registration and Object Recognition Using Affine Invariants and Convex Hulls.* IEEE Transitional on Image Processing, vol. 8, no. 7 (June 1999).

[6] I. Weiss. *Noise Resistant Invariants of Curves.* Geometric Invariance in Computer Vision, Cambridge, MIT Press (1992), 135–156.

[7] Y. Lamdan, J.T. Schwartz, H.J. Wolfson. *Object Recognition by Affine Invariant Matching.* Computer Vision and Pattern Recognition, (June 1988), 335–344.

[8] E.J. Wilczynski. *Projective Differential Geometry of Curves and Ruled Surfaces.* B. G. Teubner, Leipzig (1906).

[9] I. Weiss. *Projective Invariants of Shapes.* Proc. Image Understanding Workshop, (1988), 1125–1134.

[10] W.S. Ibrahim Ali, F.S. Cohen. *Registering Coronal Histological 2-D Sections of a Rat Brain with Coronal Sections of a 3-D Brain Atlas Using Geometric Curve Invariants and B-spline Representanion.* IEEE Transaction on Medial Imaging, vol. 17, no. 6 (December 1998).

[11] A.M. Bruckstein, E. Rivlin, I. Weiss. *Scale-space Semi Local Invariants*. Image and Vision Computing, vol. 15 (1997), 335–344.

[12] F. Mokhtarian, S. Abbasi. *Shape Similarity Under Affine Transform*. Pattern Recognition, vol. 35 (2002), 31–41.

Data on-line Monitoring and Production in High-Energy Physics Experiments

Antonín Král

3rd year of PGS, email: `antonin.kral@cern.ch`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Miroslav Virius, Katedra softwarového inženýrství v ekonomii, Fakulta jaderná a fyzikálně inženýrská, ČVUT

Abstract. The goal of this paper is to describe techniques used in data production in particle physics experiments. To pin-point some issues and describe possible solutions. We will work in context of two experiments – COMPASS and PHENIX.

Abstrakt. Cílem toho příspěvku je poskytnout přehled o úkolech, problémech a jejich řešeních, se kterými se můžeme setkat při práci na pořizování a především zpracování dat získaných při experimentech částicové fyziky, jakými jsou COMPASS a PHENIX.

1 Introduction

Paper is divided into two sections. The first one describes system used for on-line monitoring of the COMPASS experiment and extensions, which have been implemented by the author of the paper.

The second part of the paper describes data management system used in COMPASS for taking care of raw data. This system is glue between data taking and data production.

The last part of the paper deals with production of the measured data. Which is crucial part of every experiment, because thats the moment when you are trying to grub useful data out from what you have measured during data taking runs.

COMPASS (COmmon Muon Proton Apparatus for Structure and Spectroscopy) [8][2] is a high-energy physics experiment at the Super Proton Synchrotron (SPS) at CERN in Geneva, Switzerland. The purpose of this experiment is to study the hadron structure and hadron spectrum using high intensity muon and hadron beams.

PHENIX, the Pioneering High Energy Nuclear Interaction eXperiment [10][3], is an exploratory experiment for the investigation of high energy collisions of heavy ions and protons. PHENIX is designed specifically to measure direct probes of the collisions such as electrons, muons, and photons. The primary goal of PHENIX is to discover and study a new state of matter called the Quark-Gluon Plasma. Experiment is running at the Relativistic Heavy Ion Collider (RHIC) in Brookhaven National Laboratory, NY, USA.

2 COMPASS On-line Monitoring System

COOOL[6][5] is the monitoring program for the COMPASS experiment. It provides on-line or off-line access to the data taken in COMPASS. These data are decoded and

processed, then they are shown typically as histograms or two-variables graphs or can be saved to a TTree structure for later processing using ROOT. Actually ROOT [7][4] is used as a framework for the whole application.

The whole experiment contains over 200 detectors, which are all monitored by COOOL during a run. These detectors are mainly trackers (scintillating fibers, MWPC's, drift chambers, micromegas, silicon detectors, GEM's, straws and drift tubes), but there are also several planes of hodoscopes, 2 hadronic calorimeters and a RICH. COOOL has defined an object class (COOOL is written in C++) for each type or group of similar detectors. The scheme of the interesting part of object hierarchy is shown at Fig. 1.

2.1   Extending COOOL

Main task of the author was in extending COOOL to support new types of detectors added for hadron experiments. Because the whole application is designed with object-oriented programming paradigm on mind, extension is mainly question of writing new classes for the added detector. Part of the hierarchy is shown on Fig. 1.
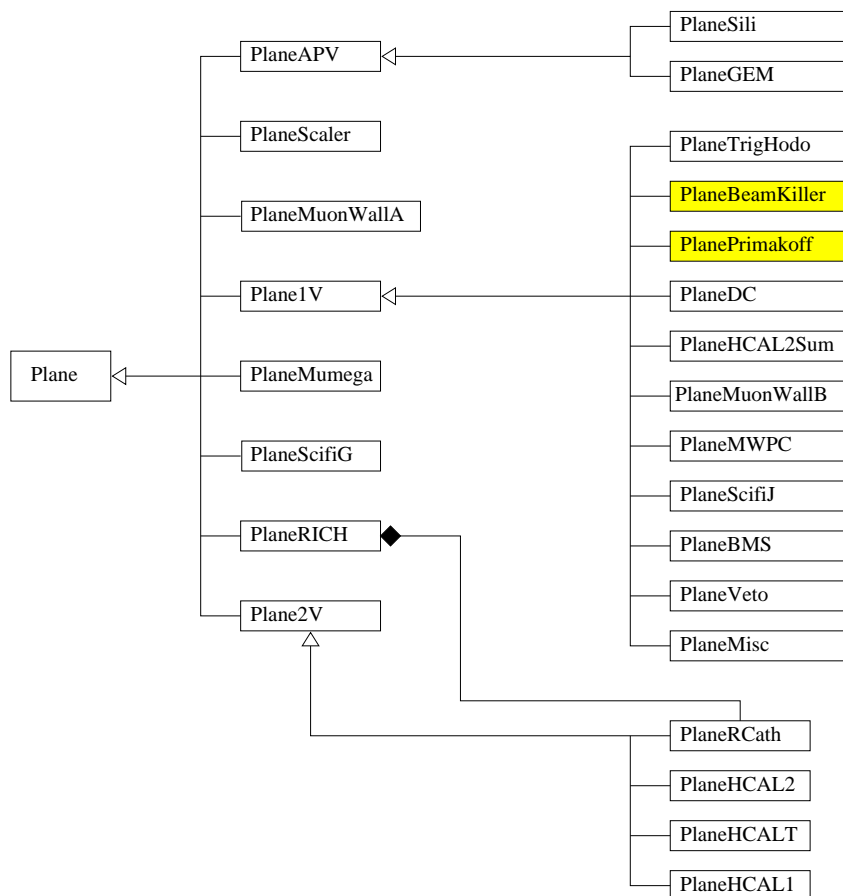
Figure 1: Part of COOOL's object hierarchy – the Planes subtree, colored classes are newly added ones

The benefit is clearly visible on source code fragment (Fig. 2) which shows implemen-

tation of time correlation among all involved Beam Killers. Operational staff on shifts are able to access output of this correlation method through graphical user interface (GUI) of COOOL as is how on Fig. 3.

```cpp
void GroupBeamKiller::EndEvent(const CS::DaqEvent &event) {
  fRateCounter++;
  Group::EndEvent();

  if (thr_flag) TThread::Lock();

  register unsigned int histIndex = 0;
  register unsigned int numPlanes = fPlanes.size();
  if (numPlanes>0) for (register unsigned int i=0; i<numPlanes-1; i++) {
    for (register unsigned int j=i+1; j<numPlanes; j++) {
      register int numTime1 =
        (dynamic_cast<const Plane1V*>(fPlanes[i]))->
                                      GetTimeVariable().GetNvalues();
      register int numTime2 =
        (dynamic_cast<const Plane1V*>(fPlanes[j]))->
                                      GetTimeVariable().GetNvalues();

      register float* valuesTime1 =
        (dynamic_cast<const Plane1V*>(fPlanes[i]))->GetTimeValues();
      register float* valuesTime2 =
        (dynamic_cast<const Plane1V*>(fPlanes[j]))->GetTimeValues();

      for (register int t1=0; t1 < numTime1 ; t1++)
        for (register int t2=0; t2 < numTime2 ; t2++) {
          (dynamic_cast<TH1F_Ref*>(fHistList[histIndex]))->
              Fill(valuesTime2[t2]-valuesTime1[t1]);
        }
      histIndex++;
    }
  }

  if (thr_flag) TThread::UnLock();
}
```

Figure 2: Fragment of COOOL's source code performing time correlation between pairs of Beam Killers

2.2 Conclusion

On-line monitoring could significantly increase quality of measured data, because operation staff is able to check quality of currently taken data. If quality is low, or something
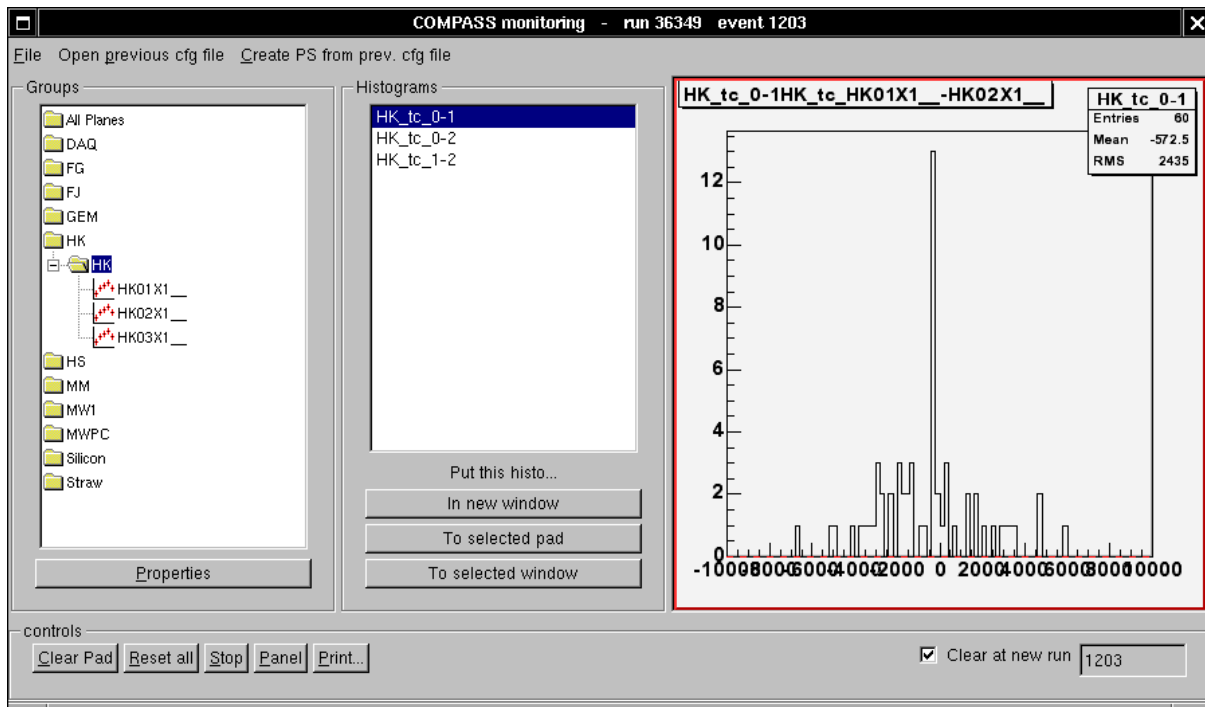
Figure 3: Main window with selected time correlation between the first and the second Beam Killer

is out of the limits, they could restart data taking, start recalibration of the apparatus or particular detector, discuss quality of the beam with SPS operations or simply mark run as not very good and thus to be removed from later processing / production.

Furthermore, extensibility of the system is very crucial as well. So we believe that system written under object oriented design with clear separation of layers dedicated for different purposes (like display, data processing, data receive etc.) is outright.

3 Data Acquisition and Central Data Recording in COMPASS

The COMPASS Central Data Recording (CDR) is combination of software and hardware for transferring raw physics data from data acquisition system (DAQ) to permanent semi-online storage – CASTOR[1]. System has to be able to cope with huge amount of data transfers, it is pretty common to process about 1PB (1024TB) of data each year / run.

You can get a rough overview of COMPASS DAQ from Fig. 4. Detector readout electronics feeds the CATCH modules or GeSiCA boards for GEM and silicon detectors. The data are transfered from these boards via optical connection (CERN's S-Link standard) to the DAQ computers. The data received from S-Link are buffered on PCI cards called spillbuffers.

From DAQ computers, which hosts spillbuffer cards (or readout buffers, ROBs), the data are transmitted via Gigabit Ethernet links to eventbuilder computers, which combine

APV 25: GEM
Silicon

F1–TDC: LAT/SciFi/uO
FIADC: Calorimetr
GASSIPLEX: RICH

Detector
frontends

Readout
modules

GeSiCA

CATCH

TCS

TCS

Optical S–Links

MUX

Readout buffer PCs

CPU

4x
Spill
buffer

gigETH

Gigabit Ethernet

Switch

Eventbuilder PCs

Event builder
and filter

Central
Data Recording
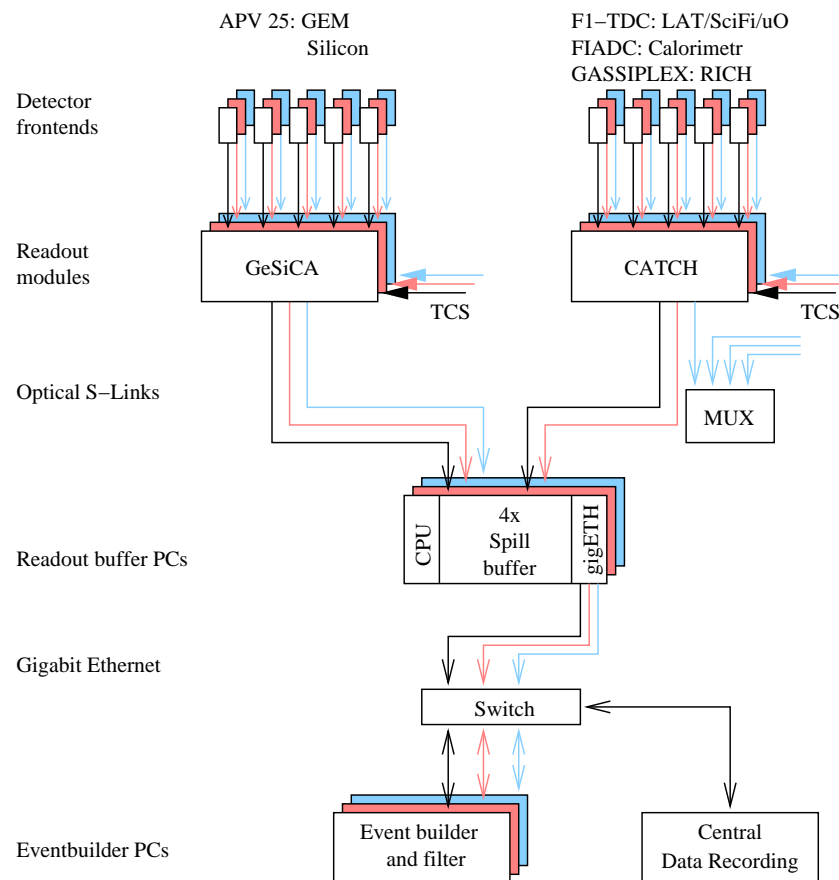
Figure 4: General architecture of the DAQ system

the sub-detector data to complete event block and transfer it to CERN's central data recording (CDR) facility. More information about whole DAQ system could be found in [20].

CDR consists of following components:

- RAW data transfer system

- meta-date generation and transfer system

- web-based information system

- CDR control and cleaning system

3.1 CDR's Basic Functionalities

The CDR system is controlled with finite automaton, which tracks state of particular chunk of take raw data. Every chunk goes through several states, each state / transition is managed with relevant daemon or script. Overview of internal states is show on Fig. 5. Detailed description of particular states could be found in [12].
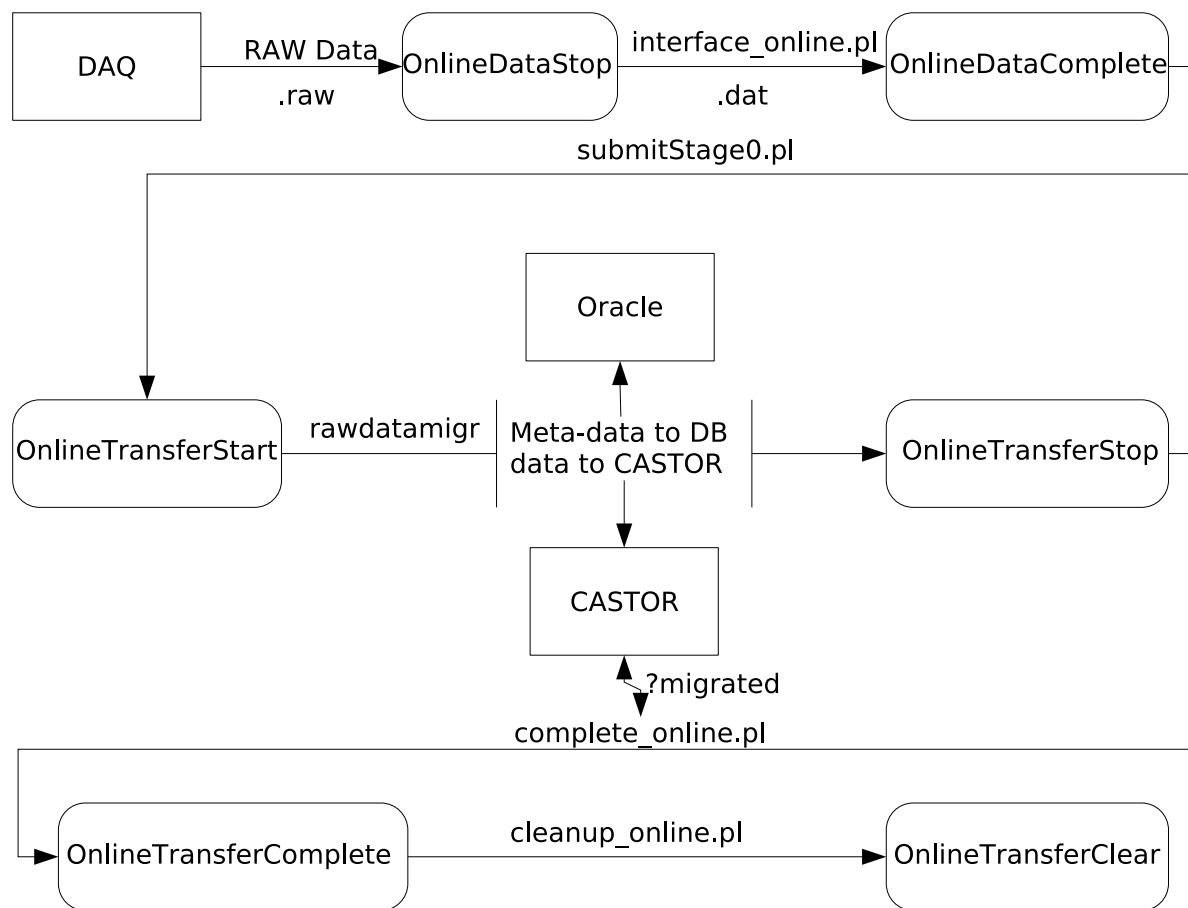
Figure 5: Internal States of CDR system

3.2   What is Really Stored?

The CDR stores couple of files and database records about each raw data chunk. These records consist mainly from following pieces:

meta-data used in research applications, because they provide information about what is included the raw data. They describe conditions under which raw data have been taken.

BKM – bookmark files contain information about whole migration process. Very useful, when something goes wrong.

log files again mainly for house-keeping tasks and debugging.

raw data the source of the real value in the experiment – data taken from particular detectors.

3.3   On-line Analysis and Monitoring

We have extended CDR system to support on-line data production. Besides only storing the data on central storage, CDR system is able to select small fraction (make sample of taken data) and submit this sample for standard production analysis. Because we process only fraction, analysis needs much smaller time to finish. Results are very useful even they have large error, because they could give very strong feedback about quality of measured data in reasonable time. And then enable operational staff to take need actions (such could be e.g. recalibration of the detector). We will see data production in more detail in the following section.

3.4   Conclusion

There is strong need for some sort of CDR subsystem in every experiment. The raw data from detectors are one of the real value in the experiment, besides experience and expertise of all involved people.

4   Data Production

Data production is another crucial part of the whole data processing chain. The main goal is to take all measured data (which are interested, which means that they have been taken under correct conditions) and run complex analysis upon this data. These analysis could contain things like beam track reconstruction, measuring of different vertexes etc.

4.1   Basic Data Production Work-flow

The author has developed framework for processing data taken in hadron part of the COMPASS experiment. This framework controls processing of every raw data chunk on the computer cluster.

Data reconstruction is CPU very intensive application, but is example of problem which could be parallelized in very efficient way. Basically every chunk of raw data could be processed separately and at the end dedicated process merges all partial results into one. Controlling production is not hard task, at least on the first sight. There is couple things which should stay under control for every moment of production:

- careful selection of chunks to be processed, to be able to deliver results as soon as possible, when results are aggregated per run.

- optimization of accesses to data storage (which is typically build on tape, where access times are significant).

- system needs to be aware of possible errors which could rise during data reconstruction, and should be able to reprocess data with different settings (after human intervention).

We have faced very similar problems as in other distributed systems, where is scheduling of resources needed, like [19].

Reconstruction itself is done with analysis software which is based on ROOT. The output files are called mDST (mini DST) and they are used by physicists during their work on fulfilling experiment's goals.

We are currently responsible for hadron as well as muon production in COMPASS. We have been involved in data production in PHENIX experiment as well. The author has proposed several strategies for improving throughput of the production. PHENIX experiment is smaller, in comparison with COMPASS, as one could see from Fig. 6, but principles behind the scene are every similar for both experiments. More details about production in PHENIX are in [9].

4.2   Conclusion

We have successfully implemented system for hadron production, we are running muon production at COMPASS and we have proposed optimization for PHENIX experiment. More information could be found in [14][16][15] and [9].

5   Conclusion and Further Work

We have described several parts of the data handling and processing chain in typical experiment. All described systems are in operation state in COMPASS or PHENIX experiments.

Author wants to focus more on networking part of the whole system in the future. One of the open questions is optimization of data transfers between distant locations as we could see in Grid based applications [11].

References

[1] Cern advanced storage management project. http://castor.web.cern.ch/castor/.
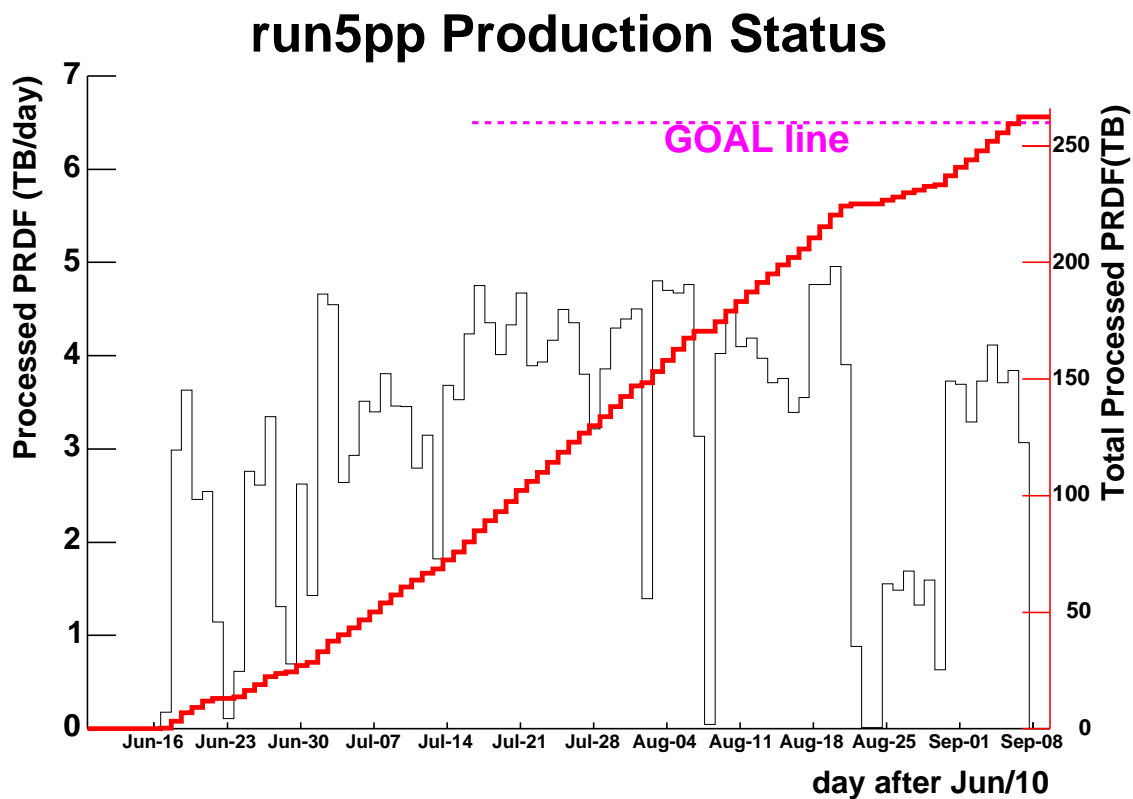
Figure 6: Data production speed (left axis) and total size of processed data (right axis). The total size of PHENIX raw data (260TB) is show as "Goal".

[2] Compass home page. http://wwwcompass.cern.ch.

[3] Phenix homepage. http://www.phenix.bnl.gov.

[4] Root – an object-oriented data analysis framework. http://root.cern.ch.

[5] C. Bernet. *Coool – compass object oriented online.*
     http://cbernet.home.cern.ch/cbernet/Coool/.

[6] C. Bernet. *Caractérisation des Micromégas et mesure de la polarisation des gluons
     sur COMPASS.* PhD thesis, Université Paris 7, (2004).

[7] R. Brun and F. Rademakers. *Root: An object oriented data analysis framework.* In
     'Nucl. Instr. Meth.', volume A389, (1997).

[8] G. Baum et al. *Compass: A proposal for a common muon and proton aparatus for
     structure and scpectroscopy.* In 'CERN-SPSL-96-14', (1996).

[9] H. Torri et al. *Mass data processing for rhic-phenix experiment.* In 'RIKEN Accel.
     Prog. Rep.', volume 38, (2005).

[10] K. Adcox. et al. In 'Nucl. Instr. and Meth.', volume A499, (2003).

[11] A. Král. *Introduction to grids.* In 'SpinFEST 2005, RIKEN, Wako', (2005).

[12] A. Král and T. Liška. *The cdr subsystem of the compass experiment.* In 'Czech. J.
     Phys. 55', (2004).

[13] A. Král and T. Liška. *Compass: New hadron run in view of coool.* Czech. J. Phys.
     55, (2004).

[14] A. Král and T. Liška. *Massive data production on distributed computation environ-
     ment.* In 'SpinFEST Nikko Workshop, RIKEN, Wako, Japan', (2005).

[15] A. Král and T. Liška. *Massive experimental data production - compass, run 2004
     results.* In 'SPIN 2005, Prague', (2005).

[16] A. Král and T. Liška. *Massive experimental data production on clusters.* In 'SPIN
     2005, Prague', (2005).

[17] A. Král, T. Liška, and M. Virius. *Compass: Počítače ve světě fyzikálních experi-
     mentů.* In 'Tvorba software', (2005).

[18] A. Král, T. Liška, and M. Virius. *Experiment compass a počítače.* In 'Československý
     časopis pro fyziku', (2005).

[19] A. Král and S. Ubik. *Scheduling and processing of performance monitoring tests.*
     Technical report, CESNET, (2004).

[20] L. Schmitt et al. *The daq in the compass experiment.* In 'IEEE Tran. Nucl. Sci.
     Proceedings of IEEE Realtime 2003 Conference', (2003).

Hodnocení radiogramů kostí ruky

Miroslav Krhounek

1. ročník PGS, email: krhounek@km1.fjfi.cvut.cz
Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská, ČVUT
školitel: Jan Flusser*, ÚTIA, AVČR

Abstract. This paper is a basic introduction / overview of possibilities to determine main features and characteristic structures of X-ray images of children's hands suitable for assessment of bone age. Proposed method is automatic reading of X-rays (minimizing the needs for user interaction); finger axes are found by rotation and summing the profiles, and they represent the information about hand rotation on the image and about ends of fingers. With proper detection it is possible to determine the position of thumb and thereby possible mirror swap (next analyses are based on standardized position - hand with thumb on the right). Preprocessing is important for next step, which is segmentation of the bones by ASM (Active Shape Models) - a local method that needs for its initialization a first approximation of the position of the bone determined.

Abstrakt. Tento příspěvek je úvodem do problematiky a přehledem možností jak určit základní vlastnosti a charakteristické struktury na rentgenových snímcích dětských rukou, vhodných ke stanovení kostního věku. Navrhovaná metoda automaticky zpracovává RTG snímky (minimalizuje spolupráci s uživatelem); osy prstů se hledají pomocí rotace a sloupcových součtů. Tím se získá informace o natočení ruky na rentgenu a o koncích prstů. Při správné detekci lze odvodit polohu palce a tím i případné zrcadlové překlopení (další analýzy jsou založené na standardizované poloze ruky s palcem vpravo). Předzpracování je důležité pro další krok, kterým je segmentace kostí pomocí ASM (Active Shape Models) - lokální metodou, která pro svou inicializaci potřebuje předpokládanou polohu hledané kosti.

1 Úvod

Důležitým faktorem v auxologii je kostní věk dávající informaci o zralosti kostry. Vychází se z metody TW3 (inovace metody TW2 - Tannera, Whitehouse et al. 1975) hodnocení kostního věku, která posuzuje velikostní i tvarový vývoj 20 kostí ruky a distální části předloktí. Kostní věk se využívá k přesné predikci dospělé výšky, při stanovování růstové diagnózy nebo při kontrole léčby pacientů s chronickými onemocněními.

1.1 Cíl práce

Cílem práce je vytvoření nástroje pro počítačovou podporu odečítání kostního věku. K dosažení tohoto cíle je potřeba řešit několik na sebe navazujících úloh:

1. segmentace kostí ruky

2. klasifikace jednotlivých kostí (TW3 – Point scoring system)

---

*Školitel specialista: MUDr. Ing. Jan Vejvalka, 2. LF UK
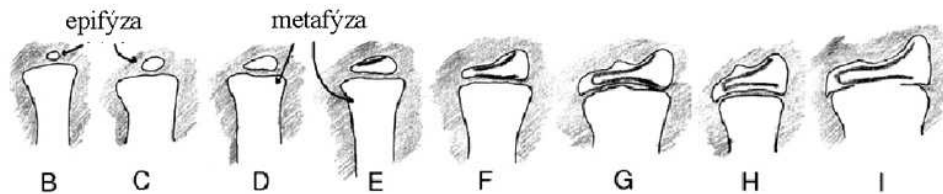
Obrázek 1: TW3 stupně zralosti pro Radius

3. z ohodnocených kostí spočítat kostní věk

Pro celkový úspěch správného určení kostního věku je základem segmentace potřebných kostí na RTG snímku. Bude-li segmentace provedena nepřesně, nedá se očekávat ani přesné zařazení do vývojových stádií, protože jednotlivá stádia se na RTG snímku někdy liší pouze v detailech.

Je požadována i velká robustnost postupu vzhledem ke vstupním RTG snímkům. Ty mohou být libovolně natočené, různě kvalitně nasnímané (naskenované na obyčejném deskovém skeneru umožňujícím skenovat průhledné materiály; naskenované na profesionálním skeneru určeném speciálně pro radiogramy; fotografované digitálním fotoaparátem aj.) a proto se může lišit jak rozlišení jednotlivých obrázků, tak úroveň jasu, kontrastu a šumu. Hodnoty jasu a kontrastu mohou být na jednotlivých RTG snímcích rozdílné nejen vlivem použité metody převodu do digitální podoby, ale vzniknou např. i samotným rentgenováním na různých pracovištích.

Po provedené segmentaci kostí lze klasifikovat jednotlivé kosti pomocí různých metod pro hodnocení kostního věku. Lze též provádět různá další kvantitativní hodnocení např. vztahů kostí na ruce, poměru délky a šířky kostí v různých vývojových etapách, délky mezi prsty apod. Takové vztahy mohou mít biologický / medicínský význam při hodnocení stavu kostního zrání jedince.

1.2 Segmentace

První úloha není triviální, jak by se mohlo na první pohled zdát. Pro vlastní segmentaci kostí použiji metodu ASM (Active Shape Models), která je lokální a pro její inicializaci je potřeba znát přibližnou polohu kostí na RTG snímku. Segmentaci lze rozdělit na podúlohy:

1. určení základních vlastností a charakteristických struktur na RTG

2. lokální segmentace kostí pomocí ASM

3. vyhodnocení kvality segmentace

1.3 Active Shape Models

Metoda ASM je založená na vytvoření statistického modelu z trénovací množiny pro body popisující hledaný objekt. Z tohoto modelu se pak vytváří podobné tvary k segmentaci

cílového tvaru v obrázku.

Jinými slovy, chceme segmentovat určitý objekt/třídu objektů (v našem případě určitou kost ruky na RTG snímku) a přitom libovolné dva stejné objekty nejsou naprosto identické, ale zahrnují určité množství variabilit, ve kterých se liší. Trénovací množinu vytvoříme tak, aby obsahovala pokud možno co nejvíce variabilit daného objektu. Každý objekt trénovací množiny pak popíšeme množinou bodů. Body umístíme na každém objektu stejným způsobem. Většinou jimi popisujeme hlavní charakteristiky objektu – hranice, důležité hrany, křivosti aj. V dalším kroku množiny bodů srovnáme tak, aby se daly spočítat průměrné pozice bodů a jejich hlavní módy variací. Srovnáním objektů na sebe dosáhneme toho, že pozice ekvivalentních bodů v různých obrázcích trénovací množiny můžeme porovnávat jednoduše zkoumáním jejich souřadnic.

Model objektu se skládá z průměrného objektu (průměrné pozice popisujících bodů) a několika vektorů popisujících módy variací. Z tohoto modelu můžeme generovat nové podobné tvary objektů, které statisticky vychází z tvarů v trénovací množině, ale nevyskytují se v ní. Segmentace objektu na novém obrázku pak probíhá iterativním způsobem. V prvním kroku je potřeba umístit model objektu na co nejbližší pozici hledaného objektu. V dalších krocích se pak iterativně zpřesňuje generování nových tvarů objektu a jejich umístění, až do určité přesnosti konvergence (kdy se dva po sobě nalezené tvary s určitou přesností nemění) nebo do předem určeného počtu iterací.

Metodu ASM jsem vybral v diplomové práci především z důvodů využití znalosti hledaného objektu – kosti ruky, která ale není pokaždé přesně stejná a může se lišit jak během svého vývoje (kost čtyřletého dítěte je jiná než stejná kost čtrnáctiletého dítěte), tak ve stejném stupni zralosti u různých jedinců. Práce pokračuje v rozšiřování databáze RGT snímků, vyhledávání vhodných snímků pro trénování a zkvalitnění trénovacích množin.

2 Předzpracování

Protože je ASM lokální metoda, potřebuje ke své inicializaci informaci o přibližné poloze hledané kosti. Proto je potřeba vstupní RTG snímek předzpracovat, určit základní vlastnosti a vyhledat charakteristické struktury, ze kterých již bude možnost odvodit přibližnou polohu kostí pro potřeby ASM. Snahou je vytvořit metodu, která nebude klást speciální požadavky na vstupní RTG snímky (ty mohou mít různou kvalitu, rozlišení, ruka může být libovolně natočená nebo zrcadlově překlopená) a která bude automatická (bez nutnosti interakce s uživatelem).

2.1 Metoda součtových profilů rotací RTG snímku

Důležitou informací je natočení ruky na RTG snímku. Úvaha, která vedla k vytvoření metody je následující: Je-li ruka ve svislé standardizované poloze, pak je 3. prst (prostředníček) rovnoběžný s osou y a v profilu sloupcového součtu vznikne v místě prstu vrchol. Pokud se bude RTG snímek otáčet, měl by se vždy prst, který se ocitne ve svislé poloze, projevit jako vrchol v příslušném součtu sloupců. Díky různé vzdálenosti prstů od osy ruky se vrcholy zobrazí v různé vzdálenosti od středu profilu. Z nalezení lokálních maxim pro jednotlivé prsty se určí vzdálenost od středu profilu a příslušné otočení. Z toho se
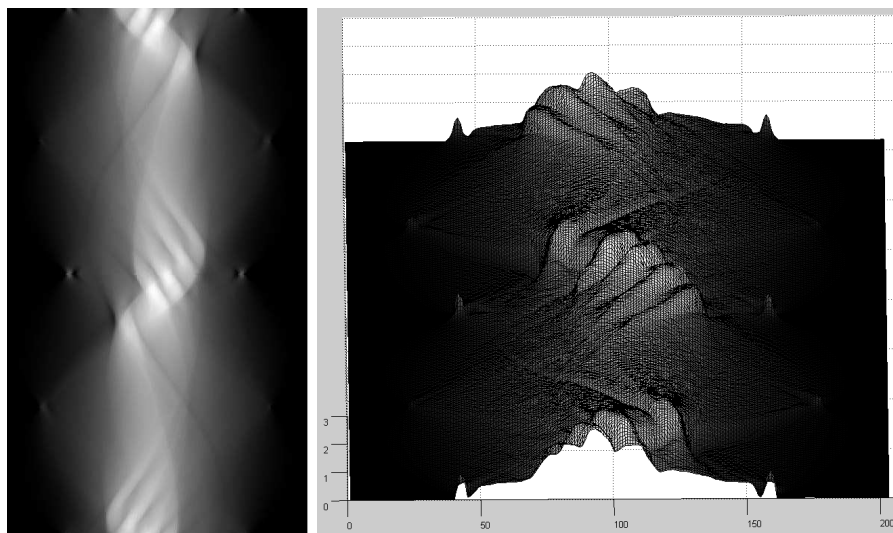
Obrázek 2: Matice součtových profilů zobrazená jako 2D šedotónový obrázek a plocha ve 3D

vypočte osa prstu. Dále můžeme analýzou lokálních minim rozpoznat oddělení prstů, což je další užitečná informace.

Implementace   Není potřeba pracovat s plným rozlišením obrázku (které může být i kolem 15 Mpix), jednak kvůli rychlosti výpočtu a dále k nadbytečné informaci. Pro toto předzpracování jsou důležité hlavní rysy kostry ruky, které vyniknou v menším měřítku. Zmenšený RTG snímek se postupně otáčí. Pro každé otočení se sečtou sloupce a výsledný profil uloží do řádku matice. Použil jsem otáčení po jednom stupni v rozsahu 1-360 stupňů. Tímto způsobem je zaručen minimálně jeden kontinuální (navazující) přechod všech prstů přes svoji svislou polohu. Počet kroků otáčení lze optimalizovat na polovinu (při úpravě matice, tak aby obsahovala kontinuální přechod prstů).

V profilu se označí lokální maxima, která by měla odpovídat prstům, resp. minima, která prsty oddělují. Předpokládám, že prst bude mít lokální maximum v profilu největší právě pro takové otočení, na kterém je rovnoběžný s osou y. Protože je 3. prst nejdelší, očekávám, že bude mít vrchol v profilu největší ze všech prstů a jeho pozice bude nejblíže středu profilu.

Poloha ruky na RTG snímku.   Z výše uvedených důvodů by mělo vyhledání globálního maxima v matici profilů určit osu 3. prstu, tedy otočení celé ruky na RTG snímku.

Problémy.   Artefakty (popisy, značky, bílé okraje, nejednotné pozadí – různé skvrny, které se mohou vyskytnout u klasických radiogramů aj.), které se na RTG snímcích objevují, zkreslují součtové profily a globální maximum, nemusí odpovídat předpokládanému 3. prstu. K odstranění tohoto problému jsem použil gaussian, který okraje snímku, kde se artefakty vyskytují, dostatečně utlumí. V součtu nebudou význačné a převládne informace, která je ve středu RTG snímku – tam se předpokládá kostra ruky, především

Obrázek 3: Automatické vyhledání os prstů - vyznačených šedě

metakarpy a minimálně první články prstů, což je pro určení osy dostatečné.

I přes odfiltrování okrajů RTG snímku neoznačuje globální maximum vždy předpokládaný 3. prst, ale nalezne se osa 2. prstu. Z pozorování se to většinou stane v případě, že je na RTG snímku menší ruka, a proto je zachycena větší část distální části předloktí (vřetení a loketní kosti). Při otočení obrázku, tak že je 2. prst svisle, se tento prst dostane do zákrytu s vřetení kostí. Zatímco pro 2. prst se do profilu nasčítá i větší část vřetení kosti (radius), osa 3. prstu směřuje mezi tyto kosti a do profilu nic navíc nenasčítá a proto bude vrchol pro 2. prst globálním maximem.

Dalším problémem je určení otočení RTG snímku o 180 stupňů, tak aby prsty směřovali vzhůru.

2.1.1 Hledání os všech prstů

Ze zobrazení matice profilů a vyznačených lokálních maxim a minim (vzhledem k řádku) je zřejmé, jak se jednotlivé prsty v této matici projevují. Při jakém otočení začne prst vytvářet vrchol v profilu, který se otáčením zvětšuje až do svého maxima, kdy je prst ve svislé poloze a opět se vrchol zmenšuje. Zároveň se vrchol v profilu posunuje, a to vytváří v matici profilů výsledné "šikmé linie". Podle směru otáčení a směru pohybu vrcholu na ose x je možné říct, v jaké poloze se ruka nachází – jestli prsty vzhůru nebo dolů.

Zobrazené šikmé linie představují jednotlivé prsty pro různá otočení RTG snímku. Jejich rozeznáním (ne vždy je linie spojitá) a oddělením od sebe získáváme informaci o jednotlivých prstech ruky. Maximum v jedné šikmé linii předpokládá určení osy prstu. Podaří-li se rozpoznat a správně přiřadit osy jednotlivým prstům, můžeme na základě svírajících úhlu a vzájemné polohy určit polohu palce a tím i případné zrcadlové překlopení RTG snímku. Navazující analýzy (převážně segmentace pomocí ASM) jsou založeny na standardizované poloze ruky s palcem vpravo.
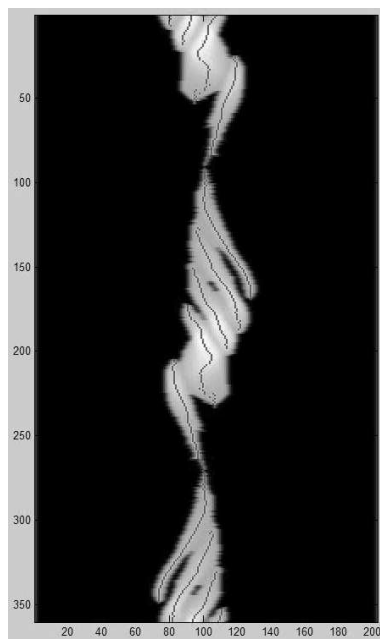
Obrázek 4: Matice součtových profilů s vyznačenými řádkovými lokálními maximy a odfiltrování hodnot menších než nejmenší maximum

2D lokální maxima První zkoušenou metodou vyhledání os všech prstů, byla analýza bodů podezřelých z 2D lokálního maxima. Vyhledal jsem lokální maxima v řádcích a zároveň sloupcích a ty jsem se pomocí shlukové analýzy snažil rozdělit do skupin pro jednotlivé prsty. V každém shluku jsem pak vybral maximum – k výpočtu předpokládané osy prstu.

   Komplikace. Body podezřelé z 2D lokálního maxima jsou vybrány převážně ze zmíněných šikmých linií. Ne vždy jsou vybrány body pro všechny prsty a pak nastává komplikace při nastavení parametrů shlukové analýzy. Není tedy zaručeno, že tato metoda vyhledá všechny osy prstů a nelze jednoduchým způsobem určit, pro které prsty se osy našly.

Analýza šikmých linií Pomocí matematické morfologie vytvořit z nespojitých šikmých linií spojité objekty uzavřením. Vzniklé objekty obarvit (rozlišit od sebe) a vyhledat ke kterému prstu odpovídá. Dále v jednotlivých objektech příslušející prstům vyhledat maxima k určení os prstů.

2.2 Současná a budoucí práce

Implementace analýzy šikmých linií a porovnání s metodou 2D lokálních maxim. Zabývat se analýzou přechodu lokálního maxima v lokální minimum v součtovém profilu, což by mohlo označovat místo konce prstu a porovnat s metodou, která vyhledává konec prstu pomocí osy prstu. Analýzou okolí osy prstu a vyhledáním oddělení prstů (z lokálních minim) segmentovat celé prsty. Ty použít k podrobnější analýze vyhledávání např. středu

kostí nebo jiných znaků pomocných ke správné inicializaci metody ASM. Použít informace o nalezených koncích prstů k předpovědi místa inicializace metody ASM pomocí vzájemných vztahů (popsaných v diplomové práci).

3 Závěr

V tomto příspěvku pro Doktorandské dny jsem nastínil problematiku, kterou se zabývám. Rozepsal jsem více první část týkající se segmentace kostí ruky, která je velice důležitou, protože pozdější kvalitativní/kvantitativní klasifikace kostí je závislá na kvalitě této segmentace. Vypsal jsem několik metod předzpracování, kterými jsem se zabýval. Tyto metody hledají základní vlastnosti a struktury na RTG snímku. Zjištěné informace slouží k inicializaci lokální metody ASM pro přesnou segmentaci kostí. V příspěvku neuvádím konkrétní výsledky statistických porovnání různých navržených metod. Zabýval jsem se zatím převážně hledáním způsobů, jak RTG snímek analyzovat a jaké informace jsou z hlediska pozdějších úloh potřebné a zajímavé. Podrobnější statistické vyhodnocení úspěšnosti různých metod na větších vzorcích dat bude v náplni další práce.

Literatura

[1] Tanner, J. M, Healy. M. J., Goldstein, H., *Assessment of skeletal maturity and prediction of adult height (TW3 Method)*, W. B. Saunders, Harcourt Publishers Limited, London, 2001

[2] Seul M., O'Gorman L., Sammon M. J., *Practical Algorithms for Image Analysis*, Cambridge University Press, 2000, ISBN 0-521-66065-3

[3] T.F. Cootes, D. Cooper, C.J. Taylor and J. Graham, *Active Shape Models - Their Training and Application*, Computer Vision and Image Understanding. Vol. 61, No. 1, Jan. 1995, pp. 38-59.

[4] Krhounek M., *Hodnocení radiogramů kostí ruky*, diplomová práce na FJFI ČVUT Praha, 2004

Integrace dat v prostředí Sémantického Webu*

Zdeňka Linková

2. ročník PGS, email: linkova@cs.cas.cz
Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská, ČVUT
školitel: Július Štuller, Ústav informatiky, Akademie věd ČR

Abstract. Data integration has been an acknowledged problem in data processing field. Its goal is usually to provide an unified view over several data sources. In case of nonmaterialized solution, it is crucial to establish relationships between provided virtual view and data in the sources. The paper deals with relationships establishment. Its approach is based on ontologies.

Abstrakt. Datová integrace je uznávaný problém v oblasti zpracování dat. Jejím cílem je obvykle poskytnout ucelený pohled na několik datových zdrojů. V případě nematerializovaného řešení je klíčové stanovení vazeb mezi poskytovaným virtuálním pohledem a daty uloženými ve zdrojích. Článek se zabývá řešením stanovení těchto vazeb. Svůj přístup zakládá na ontologiích.

1 Úvod

Dnešní svět je světem informací. Téměř vše je založeno na informacích, od pokroku v oblasti výzkumu po podnikatelský úspěch. Větší přístup k informacím umožnila také expanze World Wide Webu (WWW) – dnešní WWW obsahuje obrovské množství informací. Se stále vzrůstajícím objemem dat se však objevují nové problémy, které je třeba řešit. Nastávají potíže se správou dat a jejich zpracováním. Příčinou jsou různé datové formáty, nesourodost dat a prezentace dat způsobem, který je sice příjemný pro člověka, ale není počítačově "čitelný" a tím ztěžuje automatické zpracování. Manuální zpracování je však vzhledem k množství dat téměř nemožné. A tak, i když je požadovaná informace na WWW umístěna, může být obtížné ji najít či zpracovat a využít.

Jedním z problémů, kterými je třeba se zabývat, je tzv. *integrace dat*. Integrace dat se zabývá sloučením dat. Obvykle je jejím cílem prezentovat data pocházející z různých datových zdrojů jako jediný celek a umožnit je zpracovávat, jako by pocházela z jediného datového zdroje. Potřebu datové integrace nacházíme v mnoha oblastech práce s daty: při jejich vkládání do databáze, dotazování se nad několika tabulkami relační databáze, až po slučování více databází i jiných datových zdrojů. Například při vyhledávání informace na webu je často k odpovědi na dotaz potřeba práce s více zdroji – požadovaná informace se totiž nemusí nikde nacházet kompletní. Často je však možné na různých místech nalézt alespoň její části, ze kterých lze pak, jsou-li vhodně zkombinované, požadovanou informaci sestavit.

Datová integrace je dlouho uznávaným problémem zpracování dat. I když je předmětem mnoha výzkumů a projektů a některá data z určitých oblastí již byla integrována, stále není žádný univerzální nástroj, který by integraci dat řešil na obecné úrovni.

Různé přístupy řeší integraci na různých úrovních abstrakce, některé integrují pouze data samotná, jiné se zabývají také datovými schématy. Přístupy lze rozdělit také podle toho, v jaké formě onen ucelený přístup k datům poskytují. Jeden způsob spočívá ve vytvoření nového datového zdroje, který obsahuje všechna data z původních zdrojů. Data jsou z původních zdrojů do nového přesunuta, nebo jsou vytvořeny jejich kopie, případně jsou některá data dále nakombinována. Protože nový datový zdroj data opravdu fyzicky obsahuje, označuje se tento přístup jako *materializovaný*. Výhodou tohoto řešení je, že data jsou stále dostupná, není nutný další přístup k původním zdrojům. S tím také souvisí výhoda relativně rychlé odpovědi, nebo alespoň možnosti rychlost přístupu k datům ovlivnit. Nevýhodou mohou být však paměťové nároky a především aktuálnost dat. Při změně původních dat se integrovaná data stávají neaktuálními a je nutná změna. Tento přístup proto není vhodný pro data, která se rychle mění.

Při integraci dat na webu, a nejen tam, se stále častěji využívá *nematerializovaný* přístup. Jeho podstatou je vytvoření tzv. *virtuálního pohledu* [20] na data. Ten data fyzicky neobsahuje, data zůstávají v původních zdrojích, ale je možné k němu přistupovat a s ním pracovat, jako by data skutečně obsahoval. Data jsou pak stále aktuální. Je však nutné zajistit dostupnost zdrojů a rychlost odpovědi na dotaz je limitována rychlostí nejpomalejšího zdroje.
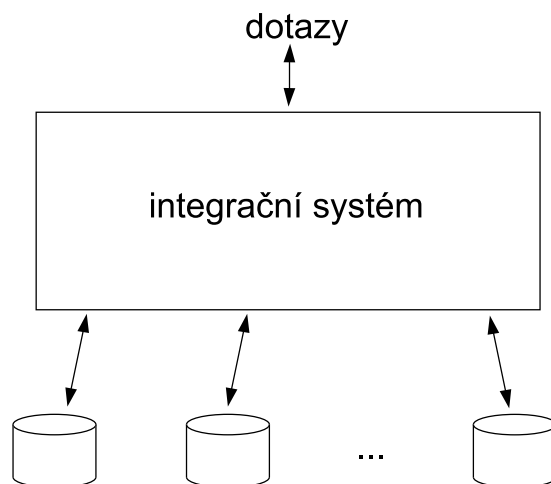
Obrázek 1: Nematerializovaný integrační systém

Aby bylo možné k integraci využít virtuální pohled, je třeba definovat jeho vazbu na fyzická data. Proto je třeba se v tomto přístupu zabývat také schématy dat. Vazba mezi pohledem a daty se pak zajistí definováním vztahů mezi jednotlivými částmi schématu pohledu a částmi schémat původních zdrojů. Stanovení takovýchto vztahů pak bývá označeno jako *mapování*. Mapování je pak využito při zpracování dotazů. Dotaz, který je kladen v prostředí – tj. jazyk a schéma – (globálního) integrovaného virtuálního pohledu, bývá pomocí mapování rozložen na části odpovídající jednotlivým (lokálním) zdrojům a přepsán do jejich prostředí. Tyto části jsou vyhodnoceny nad původními zdroji. Jejich lokální odpovědi jsou pak s pomocí mapování opět sestaveny do globální odpovědi, která je vrácena jako odpověď na kladený dotaz.

Hlavní komponenty integračního systému, založeném na nematerializované integraci, jsou zdroje se svými (lokálními) schématy, virtuální pohled s (globálním) schématem a systém mapování. Integrační systém lze tedy formalizovat jako trojici

$$I = (G, L, M),$$

kde G je globální schéma, L je množina lokálních schémat a M je systém mapování.

2   Sémantický Web

Idea *Sémantického Webu* [3], [11] vychází ze snahy umožnit zautomatizované zpracování dat. Snaží se data prezentovat v takové podobě, aby byla nejen srozumitelná pro člověka, ale také srozumitelná, čitelná a smysluplně zpracovatelná pro počítačový program. Toho chce dosáhnout doplněním sémantiky – je založen na myšlence, že spolu s daty bude definován také jejich význam.

Sémantický Web není nějaký nový samostatný web, ale je zamýšlen jako rozšíření toho současného. Je založen na několika webových technologiích a standardech, o jejichž definici usiluje W3C (WWW konsorcium) [21].

Důležitým požadavkem počítačem zpracovatelné informace je *strukturování dat*. Na webu je hlavní strukturovací technikou používání značek (tagů). Základním pilířem tvorby Sémantického Webu je v současné době značkovací jazyk XML (eXtensible Markup Language) [9]. Ovšem samotné XML a strukturování dat k počítačově čitelné informaci nestačí. Technikou, jak specifikovat význam informace je RDF (Resource Description Framework) [18]. RDF je základním nástrojem k *připojení metadat* (dat o datech). Poskytuje abstraktní model pro definici metadat a jejich použití. K vyjádření metadat modelu RDF je využívána syntaxe jazyka XML (tzv. RDF/XML). K dispozici je i rozšíření RDF zvané RDF Schema [17], s jehož pomocí lze definovat třídy a hierarchickou strukturu.

Nástrojem pro definici termínů použitých v popisu dat nebo metadat jsou *ontologie*. V kontextu webových ontologiích je ontologie soubor nebo dokument, který obsahuje formální definici termínů a vztahů mezi termíny. Technikou Sémantického Webu pro definici ontologií je jazyk OWL (Ontology Web Language) [22]. Díky používání ontologií budou moci aplikace sdílet své slovníky, což umožní jejich kooperaci.

Idea Sémantického Webu navíc zahrnuje i přidání *logiky* na web, především ve smyslu používání inferenčních pravidel. To přináší možnost odvozovat další vztahy a činit různé závěry.

Svého plného potenciálu může Sémantický Web dosáhnout jen tehdy, jestliže lidé vytvoří také programy, které budou jeho služeb využívat. Takovéto programy by měly zpracovávat obsah webových zdrojů a vzájemně kooperovat. Jejich práce bude tím efektivnější, čím více dat na webu bude počítačově zpracovatelná a čím budou přístupnější ostatní automatizované služby. Sémantický Web by tak měl poskytnout základnu pro ostatní technologie.

3   Integrace dat na Sémantickém Webu

S ohledem na množství datových zdrojů na WWW a na fakt, že některé zdroje jsou zde velmi rychle aktualizovány, je k datové integraci vhodný nematerializovaný přístup.

Klíčové je v tomto případě určení vazeb mezi jednotlivými datovými schématy. I na Sémantickém Webu je možné použít klasické přístupy [14], [4], známé jako GAV a LAV.

GAV (*Global As View*) přístup je založen na tom, že globální virtuální pohled je definován jako pohled nad lokálními zdroji. Každý element globálního schématu je tedy charakterizován jako pohled nad lokálními schématy. Tento způsob mapování vlastně systému říká jak získat data. V jednoduchém případě je pak zpracování globálních dotazů relativně snadné, každý element globálního schématu je v dotazu nahrazen pohledem, kterým je definován. Ovšem idea GAV je vhodná v případě, že je množina integrovaných zdrojů stabilní. Změna ve zdrojích, například přidání nového zdroje, může být složitá. Nový či měněný zdroj může mít vliv na definici různých elementů globálního schématu, takže je systémový návrhář nucen přepracovat schéma a uvažovat všechny zdroje znovu.

LAV (*Local As View*) spočívá v definici lokálních schémat jako pohledů definovaných nad globálním schématem. V tomto přístupu je globální schéma voleno (relativně) nezávisle na schématech zdrojů. Každý zdroj je potom charakterizován v termínech globálního schématu. Mapování tak vlastně specifikuje roli každého zdroje v globálním schématu. V případě změny nebo přidání nového zdroje nejsou vyžadovány žádné jiné změny, jen je přepracováno nebo přidáno mapování dotčeného zdroje. Na druhou stranu, je v LAV obtížné zpracování dotazů. Jediná znalost o datech globálního pohledu je pomocí pohledů reprezentující lokální zdroje a ty obsahují pouze částečnou informaci o datech. Nemusí být proto zřejmé, jak datové zdroje pro zodpovězení dotazu využít.

Protože každý z přístupů GAV a LAV má své výhody i nevýhody v různých částech procesu integrace, objevují se i projekty, které oba přístupy různě kombinují. Například GLAV (*Global Local As View*) [10] specifikuje mapování LAV i GAV.

K popisu mapování, ať už získaném přístupem GAV, nebo LAV je možné využít různých struktur. Tyto struktury se často liší projekt od projektu, některé zachycují pouze dvojice ekvivalentních konceptů, jiné pro zachycení složitějších vztahů využívají struktur složitějších. Na Sémantickém Webu je také možné takovéto přístupy ke stanovení a popisu mapování využít. Ovšem na Sémantickém Webu je k dispozici jeden mocný prostředek, který může přispět v úloze stanovení vztahů mezi schématy a nabídnout více v případě jejich zachycení. Tím prostředkem jsou ontologie.

Pojem *ontologie* [8] bývá užíván v různých souvislostech. Velmi populární definicí ontologie v informatice je: ontologie je formální, explicitní specifikace konceptualizace. Konceptualizace se vztahuje k abstraktnímu modelu světa. Ovšem konceptualizace není platná universálně, není jednoznačný přístup k tomu, jak abstraktní model světa okolo nás vytvářet. Ontologie by měly řešit problém implicitního přístupu k modelování znalostí zavedením explicitní konceptualizace.

Je množství přístupů k datové integraci, které využívají ontologií. Ty mohou být využity v různých částech integračního procesu. Na počátku mohou být ontologie využity v datových zdrojích k popisu dat. Tyto ontologie pak mohou být použity při identifikaci sémanticky korespondujících konceptů. To je zásadní při stanovení mapování.

Některé projekty, které využívají ontologií dostupných se zdroji dat, řeší integraci klasickým přístupem GAV nebo LAV, jako například [1]. V některých projektech mají ontologie i další úlohu. Například je vytvořena také tzv. globální ontologie, tj. ontologie globálního pohledu. Ta může být definována dvěma způsoby. Jednak může obsahovat slovník, který lokální zdroje sdílejí. V některých projektech obsahuje taková ontologie

základní pojmy z konkrétní domény a je obvykle mnohem obecnější než lokální ontologie [15]. Druhou možností pak je definovat globální ontologii jako výsledek sloučení ontologií lokálních.

Mnohé projekty datové integrace zůstávají u definice mapování pro stanovení vztahů mezi globálním pohledem a lokálními zdroji, např. [7]. K mapování může být využita široká škála struktur, od jednoduchých mapovacích pravidel mezi dvěma koncepty (synonyma, homonyma, disjunktní pojmy atd.), přes mapování konceptu na jedné straně k dotazu či pohledu na straně druhé [5] (jako je tomu u GAV a LAV), až po přídavné mapovací struktury (např. model referencí v [19]). Některé projekty využívají v této fázi ontologie k určení svého pojetí mapování a vlastní mapování je vlastně instancí této ontologie mapování.

Přístup, který je prezentovaný v tomto příspěvku je podobný přístupu v [6] – je také založený na slučování lokálních ontologií. Rozdíl je v tom, že ačkoli je globální ontologie výsledkem těch lokálních, oni k popisu mezi lokálním a globálním prostředím zůstávají u "tradičního" pojetí mapování, konkrétně k mapování využívají mapovací tabulku. Přístup v tomto příspěvku navrhuje v mapování využít samotnou ontologii, která vznikne jako výsledek sloučení ontologií lokálních, ontologie globální a rovněž všech známých vztahů mezi nimi. Proto je tento přístup nejvíce podobný projektům, které byly primárně určeny ke slučování ontologií, jako např. [16].

3.1 Integrace dat založená na ontologiích

Integrační systém v tomto příspěvku je založen na nematerializovaném přístupu. Globální zdroj je reprezentován virtuálním pohledem. K zajištění přístupu k datovým zdrojům (a tím vlastně k fyzickým datům) je využit jistý druh mapování. Avšak na rozdíl od mapovacích pravidel jako tvrzení zachycujících vztahy mezi elementy jednotlivých schémat je využita složitější struktura. Vychází z předpokladu, že data, která integruje jsou data pro Sémantický Web, neboli předpokladem je dostupnost ontologií, které integrovaná data popisují.

Samotná úloha integrace je transformována na vytváření ontologie integračního systému. Tato ontologie ze své podstaty by měla pokrývat ontologie všech dat, která jsou v systému využita, a mapování, na které můžeme obecně nahlížet jako na stanovení vztahů mezi daty. Tak může být tato ontologie využita k datové integraci na úrovni schémat.

Ontologie a datová schémata obecně k sobě mají blízko. Hlavním rozdílem je především jejich účel. Ontologie bývá tvořena s úmyslem definovat pojmy používané v nějaké oblasti, zatímco schéma bývá využito k modelování nějakých určitých dat. Ačkoli není nutně platné tvrzení, že lze nalézt korespondenci mezi datovým modelem a k jeho vyjádření využitými pojmy, často tomu tak bývá. Obzvláště u schémat reprezentujících sémantický datový model, nemusí být patrný žádný rozdíl a tím ani způsob jak identifikovat co je schéma a co ontologie. V ostatních případech může být ontologie využitá pro ta konkrétní data obohacena o další koncepty a vztahy, aby zachytila i datové schéma.

Předpokládejme, že na počátku úlohy jsou k dispozici dva datové zdroje S_1 a S_2, které mají být integrovány. Každý zdroj je popsán ontologií: ontologii zdroje S_1 označme O_{S_1}, ontologii zdroje S_2 označme O_{S_2}. Globální integrovaný pohled poskytovaný integračním systémem bude popsán ontologií O_G. Integrační systém, který je v kapitole 1 formalizován

jako trojice $I = (G, L, M)$, má v tomto případě reprezentaci $I = (O_G, \{O_{S_1}, O_{S_2}\}, O_I)$, kde O_I je ontologie integračního systému.

Ontologie O_I je určena k popsání mapování mezi elementy globálního pohledu a lokálních zdrojů. Mapování je stěžejní částí integračního systému a jeho stanovení, popis a vyjadřovací síla ovlivňují množství informace, které jsme schopni integračním systémem získat. O_I je zároveň ontologií všech dat v integračním systém. Z toho plyne, že pro ontologie lokálních zdrojů platí:

$$O_{S_1} \subseteq O_I \,,$$
$$O_{S_2} \subseteq O_I \,.$$

Zatímco ontologie O_{S_1} a O_{S_2} jsou dány spolu se zdroji, O_G a O_I je třeba stanovit. Popis O_G je relativně nezávislý na zdrojích. O_G obsahuje definici všech konceptů přístupných přímo pomocí globálního pohledu. Je proto úlohou designéra, který rozhodne, co bude pomocí integračního systému přístupné a v jaké formě.

Určení O_I je zásadním krokem, nejde ovšem o nijak snadnou úlohu. Protože by O_I měla pokrývat O_{S_1}, O_{S_2}, O_G i jejich vzájemné vztahy, je O_I výsledkem úlohy nazývané slučování ontologií (*ontology merging*). Při procesu slučování je několik lokálních ontologií na vstupu, na výstupu je pak jako výsledek vrácena sloučená ontologie. Proces slučování ontologií je tématem mnoha výzkumných projektů,např. [12], a je i předmětem mnoha specializovaných konferencí.

Hlavní obtíž při sémantické integraci je stanovení korespondence. I když formální definice v ontologiích jsou nejlepší specifikace termínů, které jsou v současné době k dispozici, nemohou zachytit plný význam. Často je proto při stanovení korespondence nutná jistá lidská intervence. Ovšem i v případě, že počítačové programy nejsou schopny korespondenci odvodit, mohou být využity k návrhům možných vztahů či k ověření vztahů zadaných člověkem.

Stejně jako při integraci schémat v jiných přístupech, i zde mohou vyvstat některé konflikty [2], které je třeba řešit. Obecně mohou být takové konflikty několika typů [13], například to mohou být konflikty mezi jednotlivými termíny (synonyma, homonyma atd.), schématické nesourodosti, konflikty mezi vlastními daty a metadaty atd. Ve světě ontologií není obtížné uvažovat rozdílné koncepty, protože jsou zde prostředky, jak vyjádřit vztahy mezi nimi. V ontologii má každý pojem jednoznačnou referenci. I když mohou být dva termíny ve dvou ontologiích shodně nazvány, jsou odlišitelné díky kontextu – ontologii, ve které jsou definovány. Toto je například v syntaxi XML řešeno pomocí jmenných prostorů (namespaces). V ontologiích je také možné stanovit, že dva termíny jsou ekvivalentní a umožnit tak, aby byly podle toho zpracovány.

Příklad 1. Předpokládejme obchod, který prodává notebooky od různých výrobců. Pro jednoduchost předpokládejme pouze dva výrobce: Sony a Asus. Každý z nich poskytuje data o svých produktech. Obchod by však přivítal přístup k těmto datům jako k jedinému celku, proto je nutné oba zdroje s daty integrovat.

K integraci jsou dva zdroje. Zdroj 1 obsahuje notebooky produkované firmou Sony. Jeho ontologie O_1 obsahuje pouze jedinou třídu nazvanou `notebooky_Sony`, které mají vlastnosti `Rychl_procesoru`, `Paměť` atd. Zdroj 2 obsahuje notebooky od producenta Asus. Jeho ontologie O_2 obsahuje třídu `notebooky_Asus` s vlastnostmi `RychlostProc`, `RAM` atd.

Protože integrační systém by měl poskytovat popis notebooků z různých zdrojů, globální ontologie O_G obsahuje třídu `notebooky` s vlastnostmi `Rychlost_procesoru`, `RAM` atd. Pro získání ontologie celého systému, jsou O_1, O_2, O_G a znalosti o vztazích mezi koncepty sloučeny. Výsledná O_I je následující:
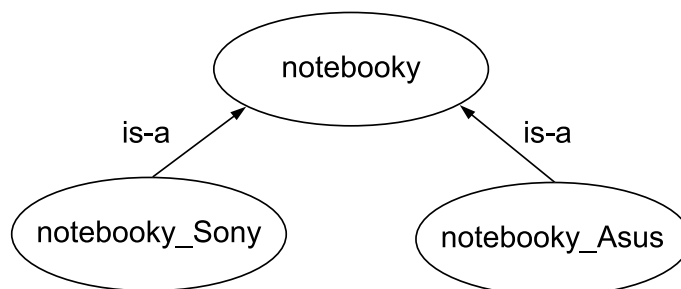
Obrázek 2: Ontologie O_I

Ontologie O_I obsahuje tři třídy: `notebooky`, `notebooky_Sony` a `notebooky_Asus`. Notebooky Sony i Asus jsou notebooky, proto je hierarchický vztah třída – podtřída mezi třídami `notebooky` a `notebooky_Sony` a mezi `notebooky` a `notebooky_Asus`. Notebooky Sony a Asus nemohou být sloučeny do jediné třídy, protože tyto třídy obsahují rozdílné notebooky. Se znalostí sémantiky jednotlivých vlastností tříd, lze podobně nalézt hierarchické vztahy vlastnost – podvlastnost, například mezi `Rychlost_procesoru` a `Rychl_procesoru`. Navíc, jsou-li na tyto vlastnosti kladeny stejná integritní omezení, je možné je sloučit a označit jako ekvivalentní. □

Uživatel integračního systému klade své dotazy v prostředí globálního pohledu (tj. schéma, jazyk atd.). Aby bylo možné tento dotaz nad lokálními zdroji vyhodnotit, je nutné jej zpracovat. V zásadě jsou dva přístupy ke zpracování globálního dotazu. V prvním je dotaz přepsán – je dekomponován na části odpovídající jednotlivým zdrojům a přepsán do příslušného lokálního prostředí. Druhý přístup se zabývá hledáním odpovědi na dotaz, kdy neklade žádné nároky na to, jak je dotaz proveden, ale jediným cílem je vyšetřit všechny možné informace s cílem najít množinu dat, která (s využitím znalostí) logicky implikuje, že jde o odpověď na dotaz.

Máme-li mapování vyjádřeno v ontologii a uvažujeme-li pouze hierarchický is-a vztah, je možné pro zpracování dotazu využít pravidlo dobře známé z objektově-orientovaného programování: potomek může zastoupit svého předka. Jestliže například hledáme všechny instance třídy T, které mají vlastnost $V = x$, je dotaz následující:

$$q := T(V = x) .$$

S využitím ontologie O_I poskytuje is-a vztah prostředek, jak dotaz přepsat s ohledem na určitý lokální zdroj. Jestliže T není konceptem lokálního schématu, je třída T v dotazu nahrazena nejbližším svým potomkem T'. Toto pravidlo je rekurzivně aplikováno, dokud není malezen kontext lokálního schématu, nebo není-li již k dispozici další podtřída – v tom případě je odpověď na (lokální) dotaz prázdná. Podobně je toto pravidlo aplikovatelné pro přepisování vlastností stanovených v dotazu.

Při druhém přístupu k zodpovídání dotazů je is-a hierarchie také klíčová. Vyjadřuje totiž, že instance libovolného uzlu je zároveň instancí uzlu v hierarchii nad ním. Na základě tohoto poznatku je možné určit, zda informace z lokálního zdroje může být odpovědí na globální dotaz.

Příklad 2. V pokračování jednoduchého příkladu integrace notebooků je ukázáno zpracování dotazů. Globální pohled poskytuje notebooky. Dotaz: vyber všechny notebooky s rychlostí procesoru 1.6 GHz, t.j.

$$q := notebooky(Rychlost\_procesoru =' 1.6') \,,$$

je zpracován následovně: `notebooky` není konceptem žádného z lokálních zdrojů, dotaz je přepsán. Třída `notebooky` má dva potomky `notebooky_Sony` ze zdroje 1 a `notebooky_Asus` ze zdroje 2. Dotaz je tedy přepsán do dvou podob:

$$q_1' := notebooky\_Sony(Rychlost\_procesoru =' 1.6') \,,$$
$$q_2' := notebooky\_Asus(Rychlost\_procesoru =' 1.6') \,.$$

Protože vlastnost `Rychlost_procesoru` není koncept zdroje 1, je dotaz q_1' dále přepsán s využitím vztahu vlastnost – podvlastnost na:

$$q_1'' := notebooky\_Sony(Rychl\_procesoru =' 1.6') \,,$$

Dotaz q_1'' je proveden nad zdrojem 1. Analogicky je přepsán dotaz q_2' a proveden nad zdrojem 2. □

Ontologie je tedy schopná zachytit jednoduchý is-a vztah. Ontologie je však mnohem silnější. Poskytuje dost silné prostředky, aby zachycovala množství různých vztahů. Aby bylo umožněno získat ze zdrojů co největší množství informace, je vhodné využít nějaký inferenční mechanismus, který umožní pracovat i se vztahy, které ontologie pokrývá, nicméně nejsou v ní přímo vyjádřeny – je však možné je z vyjádřených vztahů dále odvodit.

4 Diskuze a závěr

Řešení integrace dat navrhované v tomto příspěvku vychází z technik Sémantického Webu. Je založena na ontologiích – vychází z předpokladu, že integrovaná data jsou popsána v ontologiích, které jsou dostupné, a ontologii dále využívá jako prostředek ke stanovení vazeb mezi původními daty a poskytovaným integrovaným pohledem. Vlastní úloha integrace je převedena na úlohu slučování ontologií.

Použití ontologií (a jejich definice pomocí OWL jako standardního prostředku Sémantického Webu) v integračním systému může přinést řadu výhod. Ontologie jsou velmi silný prostředek pro zachycení vztahů mezi schématy jednotlivých zdrojů. Ontologie jako obecný prostředek, tedy na rozdíl od jiných struktur mapovacích pravidel definovaných primárně pro potřeby integrace, může být dále použita i v jiných souvislostech. Velkou výhodu použití ontologie v integraci lze nalézt při změně sytému, například při přidání nového zdroje. V takovém případě není nutná změna stávající ontologie – není tedy nutné jako ontologii nebo některou její část přepracovat, či odstranit. Původní ontologie je obohacena o nový stav změněného zdroje, či o nový zdroj, jiné změny nejsou nutné.

Literatura

[1] B. Amann, C. Beeri, I. Fundulaki, and M. Scholl. *Ontology-Based Integration of XML Web Resources*. Proceedings of the *1st Int. Semantic Web Conference (ISWC 2002)*, (2002), 117–131.

[2] S. Bergamaschi, S. Castano, M. Vincini, and D. Beneventano. *Semantic integration of heterogeneous information sources. Data & Knowledge Engineering* 36 (2001), 189–210.

[3] T. Berners-Lee, J. Hendler and O. Lassila. *The Semantic Web. Scientific American*, vol. 284, 5, (2001), 35–43.

[4] A. Cali, D. Calvanese, G. De Giacomo, and M. Lenzerini. *On the Expressive Power of Data Integration Systems*. In Proceedings of the *21st Int. Conf. On Conceptual Modeling (ER 2002)*, LNCS 2503, Springer, (2002), 338–350.

[5] D. Calvanese, G. De Giacomo, and M. Lenzerini. *Ontology of integration and integration of ontologies*. In Proceedings of the *2001 Description Logic Workshop (DL 2001)*, (2001).

[6] I. F. Cruz, H. Xiao, and F. Hsu. *An Ontology-based Framework for XML Semantic Integration*. In Proceedings of the *8th Int. Database Engineering and Application Symposium (IDEAS'04)*, Coimbra, Portugal, (2004), 217–226.

[7] Z. Cui, D. Jones, and P. O'Brien. *Issues in Ontology-based Information Integration*. In Proceedings of the *IJCAI Workshop: Ontologies and information sharing*, Seattle, USA, (2001).

[8] Y. Ding, D. Fensel, M. Klein, and B. Omelayenko. *The semantic web: yet another hip?. Data & Knowledge Engineering* 41 (2002), 205–227.

[9] Extensible Markup Language (XML). `http://www.w3.org/XML/`.

[10] M. Friedman, A. Levy, and T. Millstein. *Navigational plans for data integration*. In Proceedings of the *16th Nat. Conf. On Artificial Intelligence (AAAI'99)*, AAAI Press, (1999), 67–73.

[11] M.-R. Koivunen and E. Miller. *W3C Semantic Web Activity*. In the proceedings of the *Semantic Web Kick/off Seminar*, Finland, (2001).

[12] K. Kotis, G. A. Vouros, and K. Stergiou. *Towards automatic merging of domain ontologies: The HCONE-merge approach. Web Semantics: Science, Services and Agents on the World Wide Web* 4 (2006), 60–79.

[13] C.-Y. Lee and V.-W. Soo. *The conflict detection and resolution in knowledge merging for image annotation. Information Processing and Management* 42 (2006), 1030–1055.

[14] M. Lenzerini. *Data Integration: A Theoretical Perspective.* In Proceedings of the *21st ACM SIGMOD - SIGACT - SIGART symposium on Principles of database systems,* ACM Press, (2002), 233–246.

[15] N. F. Noy. *Semantic Integration: A Survey Of Ontology-Based Approaches.* In *ACM SIGMOD Record, Special Section on Semantic Integration,* vol.33, 4, (2004), 65–70.

[16] N. F. Noy and M. A. Musen. *The PROMPT suite: Interactive tools for ontologzy merging and mapping. International Journal of Human-Computer Studies* vol. 56, 6,(2003), 983–1024.

[17] RDF Vocabulary Description Language 1.0: RDF Schema. *W3C Recommendation.* `http://www.w3.org/TR/2004/REC-rdf-schema-20040210`, February, 2004.

[18] Resource Description Framework (RDF). `http://www.w3.org/RDF/`.

[19] H. T. Uitermark, P. J. M. van Oosterom, N. J. I. Mars, and M. Molenaar. *Ontology-based integration of topographic data sets. International Journal of Applied Earth Observation and Geoinformation* 7 (2005), 97–106.

[20] J. D. Ullman. *Information integration using logical views. Theoretical Computer Science* 239 (2000), 189–210.

[21] W3C (WWW Consortium). `http://www.w3.org`.

[22] Web Ontology Language (OWL). `http://www.w3.org/2004/OWL`.

Blind Image Authentication

Babak Mahdian

3rd year of PGS, email: `skyline@centrum.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Ing. Stanislav Saic, CSc., Institute of Information Theory and Automation Academy of Sciences of the Czech Republic

Abstract. In our society digital images are a powerful and widely used medium of communication, and they have an important impact on our life. In the recent years, due to the advent of low-cost and high-performance computers, more friendly human-computer interface, and the availability of many powerful and easy to control image processing and editing software packages, it has become possible, with a relative ease, to create image forgeries, which are by naked eye indistinguishable from authentic photographs. Therefore, there exists a strong need for a reliable tamper detection system for digital images. In this work we address the problem of passive-blind image authentication. Blind image authentication techniques use only the image function, assuming no explicit prior knowledge about the analyzed image. We introduce various methods to detect different types of image forgeries. In concrete, we will concern with detection of resampling, duplicated image regions, double JPEG compression and inconsistence noise patterns. All methods assume the absence of any digital watermark or signatures in the image. Our main aim is to make the successful use of image forgery as harder as it is possible.

Abstrakt. V dnešní době jsou digitální snímky silným a široce rozšířeným komunikačním médiem a mají zásadní vliv na náš život. V posledních letech příchod cenově dostupných a vysoce výkonných osobních pocítačů, taktéž více přátelské rozhraní mezi clověkem a počítačem, a schopností softwarových balíku ke zpracování obrazu způsobují jednoduchou tvorbu falešných obrazů. Tyto obrazy často bývají nerozeznatelné od pravých fotografií. Proto existence spolehlivého systému pro ověření pravosti obrazu je v dnešní době velmi potřebná. V tomto příspěvku se budeme zabývat problémem pasivní ci slepé autentičnosti obrazu. Tato technika ověřování pravostí ke své práci nepotřebuje mít žádné apriorní znalostí o snímku, používá jen obrazovou funkci. V této práci představíme krátce několik pasivních způsobů detekce falešných snímku. Konkrétně se budeme zabývat detekcí převzorkování, duplikovaných oblastí, dvojité JPEG kompresí a inkonzistence šumu v obrazu. Všechny tyto metody pracují bez použití watermarku či digitálního podpisu. Našim hlavním cílem je minimalizovat možnost úspěšného používání zfalšovaných digitálních snímků.

1 Introduction

In our society digital images are a powerful and widely used medium of communication, containing huge amount of information of many types. They are a compact and easy way in which to represent the world that surrounds us. Specially, we saw in the past decade a remarkable growth in our ability to capture, manipulate, and distribute digital images. In the present day, there exist many powerful public available image processing software packages, which allow users easily to change the information represented by an image

without leaving any obvious traces of tampering. So, the question is, how much can we trust a photograph, which is not obtained from a secure source.

At present, images have an essential impact on our society and play a crucial role in most people lives. Without a doubt, their trustworthiness plays a significant role in many social areas. For instance, every day newspapers and magazines are dependent on digital images. Photographs play an important role also in courtrooms, where they are used as evidence. So, it is obvious that, there is a strong need for a reliable digital forgery detection system. Such a system will be very useful among others in the following areas: forensic investigation, criminal investigation, insurance processing, surveillance systems, intelligence services and journalism. Such a system can give an advice, how much to trust a digital image.

Controversial tampered photographs have been a part of history of photography since ever. It is almost as old as photography itself. It probably began soon after Niepce Nicephore, who in 1826 created the first permanent photographic image. This type of photo tampering needed special technical equipment and experts to do the photo manipulation. But, in recent years, due to the availability of many powerful and easy to control image processing and editing software packages, such as Adobe Photoshop, and more user friendly human computer interface (HCI), digital images become more easily to manipulate and edit than ever before. Today, even non-professional users are able using average home computers to edit a digital image to the point where it is visually un-noticeable. It is possible to change the information represented by an image and create forgeries without leaving any obvious traces of tampering. It is not easy to look at an image and decide if it is real or not.

Figure 1: An example of a historical photo tampering. The man (Vlado Clementis) first from the left in the left hand picture was one of the victims of purges following the coup of 1948.

In the present day, we face the problem of digital forgery even in scientific literature. For instance, the Journal of Cell Biology, a premier academic journal, estimates that

around 25 percent of manuscripts accepted for publication contain at least one image that has been inappropriately manipulated. In many cases, the author is only trying to clean the background and his changes do not affect the scientific meaning of the results. But, there exist also articles containing tampered images in which the results were significantly affected and changed. One of the recent famous cases of digital image forgeries in scientific literature was the South Korean scientist Hwang Woo-suk's claim that he cloned stem cells. The detection of forgery in this case did not need specialized software.

Another recent famous case of digital image manipulation is a photograph taken during the 2003 Iraq war. Brian Walski, who was covering the war in Iraq for the Los Angeles Times, combined two of his Iraqi photographs into one to improve the composition and to create a more interesting image. This image showing an armed British soldier and Iraqi civilians under hostile fire in Basra was widely published. The soldier is gesturing at the civilians and urging them to seek cover. The standing man holding a young child in his arms seems to look at the soldier imploringly. It is the kind of picture that wins a Pulitzer. The tampering was discovered by an editor at The Hartford Courant, who noticed that some background people appeared twice in the photograph. It was ended with the firing of the photographer.

Figure 2: Brian Walski, a Los Angeles Times reporter, combined two photographs into one used on the newspaper's front page.

1.1 Objectives

As was mentioned, we concern with the detection of tamper in digital images. It should be noted that our goal is not to stop the forgery. This is impossible. Nowadays, many powerful image processing tools allow users to play with digital images and change the information represented by them. A capable falsifier with enough experience can always come up with an almost perfect forgery, which will be not detectable by proposed methods. Our main aim is to make the detection of forgery more reliable by developing new and more sophisticated methods. By this way we can significantly reveal many cases of tampering and more successfully distinguish between originals and fakes. Without trustworthy of photographs, we will not be able to know what in our world is real?

In this work, we will mainly introduce the state of the art, and will survey the work done in this field. Our focus will be on surveying available methods, which use only the image function and work in the absence of any digital watermark or signatures. These methods are called passive or blind and are regarded as the new direction.

Furthermore, we will propose new and more reliable methods to detect different types of image tampering with no explicit prior knowledge of the image. This part will employ the largest focus of our future work and will consist of the following sections:

- Detection of resampling. In this part we introduce a method for resampling detection working with no apriori information about the image. The method is based on detection and analyzing non-randomness features, which are brought into the signal by the interpolation step of resampling.

- Detection of duplicated regions. This part proposes a method for detection of duplicated regions of image based on moment blur invariants [2, 3]. The method works with no apriori information.

- Detection of double JPEG compression. In this part we propose an automatic method for detection of double JPEG compression based on detection of specific artifacts of double JPEG compression.

- Detection of inconsistent noise patterns. In this work we will propose an automatic method for detection of inconsistent noise patterns. The method will be based on estimation and comparison of local noise patterns and on automatic detection of inconsistence among them.

2   Tamper Detection in Digital Images

In the last chapter we explained the importance of tamper detection in images. It is obvious that there many ways how to manipulate and alter digital images. The following pages provide an attempt of categorization of the image tampering and mainly take a look at state of the art approaches for tamper detection in digital images.

2.1   Digital Image Tampering Categories

In order to divide the tampering into different groups, we will use the categorization proposed by Hany Farid [1]. But, please note that, most of high quality image forgeries use the below presented types simultaneously. Also note that, not all of the following types of image manipulation can be denoted as illegitimate. This depends on the concrete case and the ethics of image editing. For instance, some image manipulations for an image, which is used in a magazine, could be unacceptable in courtrooms. The line between improving an image and altering it has different position and width in different cases.

2.1.1   Composited

As pointed out in [1], compositing is probably the most common form of image tampering. The main idea is to combine several (mostly two) images to create one. The result of

such a forgery mainly depend on how well image components are matched in terms of size, pose, color, quality, and lighting. Many digital image processing tools propose a user-friendly environment to create such forgeries.

2.1.2 Morphed

Morphing is a digital technique that transforms the source image into the target image. Many image processing tools propose morphing operations. For an example of morphing sequence.

Figure 3: An example of a morphing sequence.

2.1.3 Retouched

This category mostly includes operations such copying and pasting of regions within the same image, air-brushing, blurring, painting, etc.

2.1.4 Enhancing

This is a class of more global digital image techniques including for example: contrast adjustment, blurring or sharpening. Mostly, this type of image manipulation does not fundamentally change the information represented by the image. But, using image enhancing, for instance, we are able obscure or exaggerate some image details.

2.1.5 Computer Graphics

This type of tampering refers to images that are generated using a computer and computer graphics methods. In the present day, the quality of such images is very high and often it is not easy to determine if the image is real or computer generated. Mostly, such images are generated by constructing a 3-D polygonal model, which is augmented with color and texture, and also illuminated with one or several virtual light sources. Finally, the augmented model is rendered to create a synthetic image.

2.1.6 Painted

This technique requires a highly talented painter with good technical skills to yield real images. Images are created from a blank screen using photo editing and drawing software. This is very similar to an artist painting on traditional canvas.

2.2   Digital Forgery Detection Techniques

It is possible to divide the digital forgery detection approaches to active, and passive (blind). Active approaches could be further divided to digital watermarks [13, 7, 9], and signatures [14, 5]. Digital watermarking assumes an inserting of a digital watermark at the source side (e.g., camera) and verifying the mark integrity at the detection side. Digital watermarks are imperceptible, they are inseparable from the digital media they are embedded in, and they undergo the same transformations as the digital media itself. A major drawback of approaches based on watermarks is that the watermarks must be inserted either at the time of recording, or later by a person authorized to do so, which requires specially equipped cameras or subsequent processing of the original image. Furthermore, watermarks may degrade the image quality.

The digital signature methods assume extracting image features for generating authentication signature at the source side and verifying the image integrity by signature comparison at the detection side. Also here, the main disadvantage is the need of a fully secure trustworthy source. We need a common algorithm on both source and detection sides. Passive (blind) approach is regarded as the new direction. Here the detection of forgery does not need any extra information. This approach uses only the image function and works in the absence of any digital watermark or signatures. So, there is no need of explicit prior information about the analyzed image.

It is obvious that active approaches (digital watermarks and signatures) are not usable for image authentication in the case, when the image is from unknown or not secure source, which is, of course, in most cases. Therefore we will focus on digital forgery detection methods based on the blind approach.

2.2.1   Blind Approach

So far, only a handful of researchers have concerned with the detection of digital image forgery by a blind technique. At present, there exist only several forgery detection techniques in the literature [10, 8, 4, 12, 11, 6]. The area has an unexplored character and there is a lot to do. Because of this reason and because of the extraordinary difficulty of the problem, today, researches are trying to start with categorizing forgeries and analyze each type separately.

Each type of alteration changes the original function of the image. As it was mentioned, researchers try to categorize these changes to make a forgery detection system combined from separate analyzers. Different analyzers try to detect the specific characteristics of different categories. The aim is to have such combination of analyzers, which will make very difficult to have a forgery that gets through it.

3   Toward a Reliable Tamper Detection System for Digital Images

In the previous chapter we have briefly described different categories of digital image tampering. As was mentioned previously, the review of recent scientific literature shows that there is a strong need of new and more sophisticated blind digital forgery detection

methods. This is our main goal. With a little investment of time and skill of the falsifier, the forgery cannot be detected via existing published methods. This is due to highly limited abilities of these methods. We will propose and develop new mathematical and computational algorithms, and in some cases improve existing methods to make the successful use of forgery harder. For this purpose, we will propose following methods:

- Resampling detection,

- Duplicated image regions detection,

- Double JPEG compression detection,

- Inconsistent noise patterns detection.

The following pages take mainly a brief look at these methods and will state their main principles. In general, the detection abilities of these methods are based on identification of statistical changes of the original image associated with the corresponded tampering technique.

3.1   Detection of Resampling

As it was mentioned previously, typically when the falsifier tries to combine two images together, there is a need of cutting parts of one picture and placing them into another. To create a high quality forgery it is often necessary to use digital image operations such as resizing or rotating of these parts. These operations require resampling of a portion of image onto a new sampling lattice using an interpolation technique. The interpolation step often brings to the image specific correlations, which typically are invisible to the human eye. Detection of these correlations may signify tampering.

3.2   Duplicated Image Regions Detection

In this part we focus on the detection of a common type of digital image forgery - the duplicated image regions or the copy-move forgery. In a copy-move forgery, a part of the image is copied and pasted into another part of the same image typically with the intention to hide an object or a region of the image, for an example. Please note that copied regions are from the same image, and therefore they have compatible properties, such as noise component or color palette, with the rest of the image. It makes the use of statistical measures to find statistical incompatibilities in different parts of the image impossible.

A common used operation to overcome existing copy-move forgery detection methods and to make the detection of forgery more difficult is blurring. Thus, to identify duplicated regions and improve the detection abilities of the nowadays approaches, it is desirable to describe analyzed regions by some features invariant to presence of unknown blur. Furthermore, due to the fact that the falsifier can also use the additive noise to make the detection of the forgery more difficult, these invariants should work well also in the presence of such a noise. The simple incarnation of such features are moment invariants with respect to blur, which have been previously addressed by Flusser and Suk

[2, 3] and have found successful applications in many areas of image processing. We have already proposed a method for duplicated regions detection based on these invariants. The method has been sent to Forensic Science International, an international journal pertaining to forensic sciences. In this work, the similarities among invariant representation of image regions are analyzed in a space created by PCT and subsequent K-d trees.
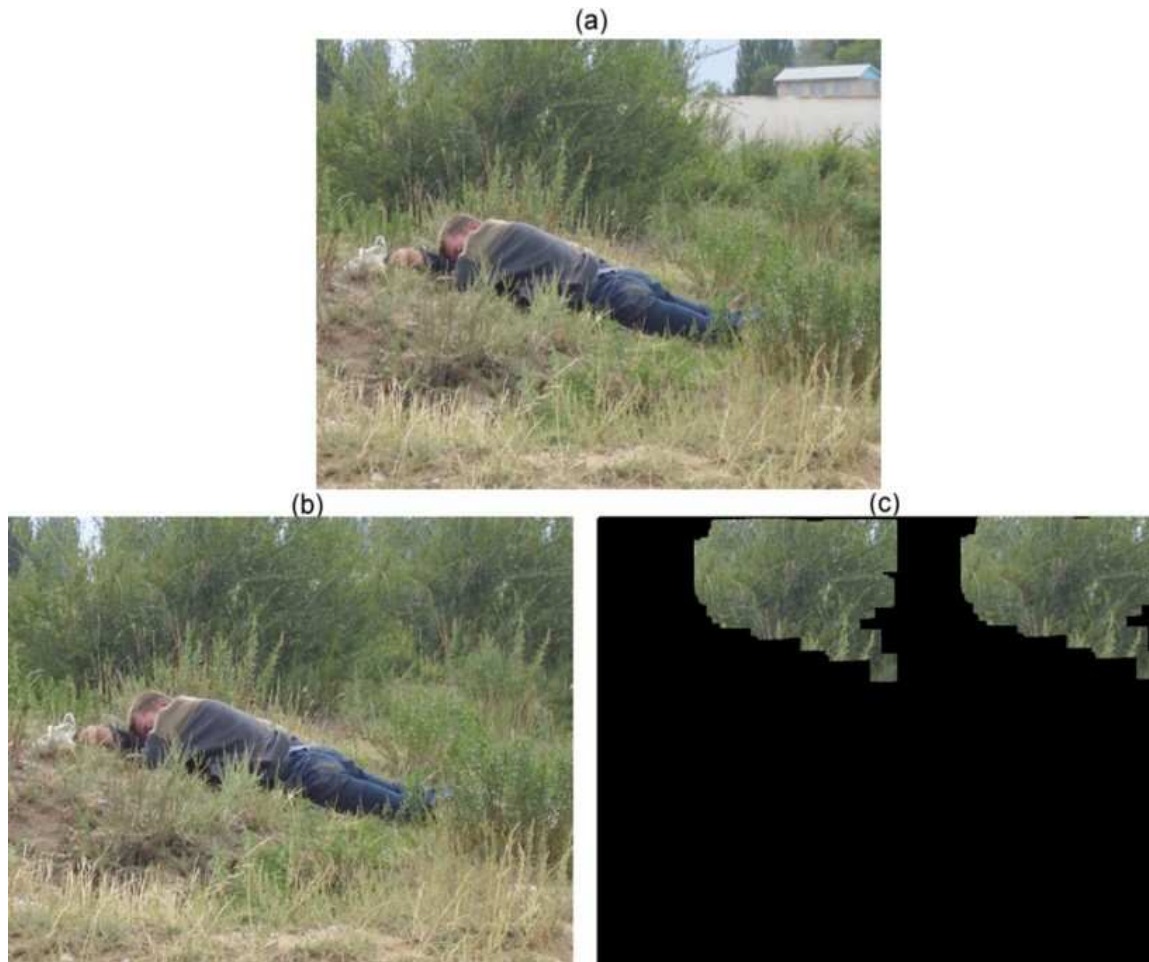
Figure 4: An example of a copy-move forgery. Shown are the original version of the test image (a), its forged version (b) and the output of our duplicated image regions method (c).

3.3   Double JPEG Compression Detection

As was mentioned previously, for altering an image, typically the image is loaded into a photo-editing software and after the changes are done, the digital image is re-saved. If the images are in the JPEG format, then the new created image will be double compressed. Double JPEG compression is imperceptible, but it introduces specific correlations between the discrete cosine transform (DCT) coefficients of image blocks. We will propose a technique to describe and detect these specific correlations. It must be noted that the

detected double JPEG compression does not necessarily means malicious tampering. Often users re-save a JPEG image by a lower quality factor to save the storage space.

3.4   Inconsistence Noise Patterns Detection

As was mentioned previously, digital images contain an inherent amount of noise, which is typically uniformly distributed across the entire image. These patterns are typically invisible to the human eye. Very often by creating digital image forgeries, the noise, which is typically in original images uniformly distributed across the entire image, becomes inconsistent. Detection of this inconsistence may signify tampering. Therefore, we will propose a method to estimate local signal-to-noise ratio. The method will be based on an additive noise model and will use blind signal-to-noise ratio estimators. Inconsistence noise patterns detection will be usable also in cases when small amounts of noise are locally added to conceal traces of tampering.

4   Conclusion

In the first part of this document we have seen that, at present, images have an essential impact on our society and play a crucial role in most people lives. In recent years, due to the availability of many powerful and easy to control image processing and editing software packages and more user friendly human computer interface (HCI), digital images become more easily to manipulate and edit than ever before. It is possible to change information represented by an image and create forgeries without leaving any obvious traces of tampering. Therefore, there exits a strong need for a reliable tamper detection system for digital images. Such a system can give an advice, how much to trust a digital image.

The middle part of this document briefly introduced the categories of the image tampering.

In the last part of document we have described our main aim and have introduced the main cognitions of methods for detection of different forms of tampering. In concrete we concerned with resampling detection, duplicated image regions detection, double JPEG compression detection and inconsistence noise patterns detection.

It is obvious that even by using and developing many more successful tamper detection methods, we will not able to detect the tamper in all tampered digital images. A capable falsifier with enough experience can always come up with something new and create an undetectable forgery. Our main aim is to make the detection of forgery more reliable by developing new and more sophisticated methods. By this way we can significantly reveal many cases of tampering and more successfully distinguish between originals and fakes. Our goal is to make the successful use of forgery as harder as it is possible.

References

[1] H. Farid. Creating and detecting doctored and virtual images: Implications to the child pornography prevention act. Technical Report TR2004-518, Department of

Computer Science, Dartmouth College, (2004).

[2] J. Flusser and T. Suk. *Degraded image analysis: An invariant approach.* IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998), 590–603.

[3] J. Flusser, T. Suk, and S. Saic. *Image features invariant with respect to blur.* **28** (1995), 1723–1732.

[4] M. Johnson and H. Farid. Exposing digital forgeries by detecting inconsistencies in lighting. In 'ACM Multimedia and Security Workshop', New York, NY, (2005).

[5] C.-S. Lu and H.-Y. M. Liao. Structural digital signature for image authentication: an incidental distortion resistant scheme. In 'ACM Multimedia Workshops', 115–118, (2000).

[6] S. Lyu. *Natural Image Statistics for Digital Image Forensics.* PhD thesis, Department of Computer Science, Dartmouth College, Hanover, NH, (2005).

[7] P. Meerwald and A. Uhl. A survey of wavelet-domain watermarking algorithms. In 'Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents III', P. W. Wong and E. J. Delp, (eds.), volume 4314, San Jose, CA, USA, (January 2001). SPIE.

[8] T.-T. Ng, S.-F. Chang, and Q. Sun. Blind detection of photomontage using higher order statistics. In 'IEEE International Symposium on Circuits and Systems (ISCAS)', Vancouver, Canada, (May 2004).

[9] A. Nikolaidis, S. Tsekeridou, A. Tefas, and V. Solachidis. A survey on watermarking application scenarios and related attacks. In 'ICIP (3)', 991–994, (2001).

[10] A. Popescu and H. Farid. Exposing digital forgeries by detecting duplicated image regions. Technical Report TR2004-515, Department of Computer Science, Dartmouth College, (2004).

[11] A. Popescu and H. Farid. *Exposing digital forgeries by detecting traces of re-sampling.* IEEE Transactions on Signal Processing **53** (2005), 758–767.

[12] A. Popescu and H. Farid. *Exposing digital forgeries in color filter array interpolated images.* IEEE Transactions on Signal Processing **53** (2005), 3948–3959.

[13] C. Rey and J.-L. Dugelay. *A survey of watermarking algorithms for image authentication.* EURASIP Journal on applied Signal Processing Volume 2002 N ° 6 - June 2002, special issue on image analysis for multimedia interactive services (2002).

[14] M. Schneider and S. F. Chang. A robust content based digital signature for image authentication. In 'IEEE International Conference on Image Processing (ICIP'96)', (1996).

Thermodynamics of Fuel Cell Membrane Transport

Ondřej Mičan

3rd year of PGS, email: `omican@gmail.com`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: František Maršík, Mathematical Institute, Faculty of Mathematics
and Physics, Charles University

Abstract. A simple diffusion type model of a PEM (proton-exchange membrane) fuel cell will be introduced. From thermodynamical analysis of the model will be obtained a relation between material properties and geometrical dimensions of the membrane. Also, coupling between water diffusion flux and electric current in the proton-exchange membrane of a PEM fuel cell will be an object of our analysis. As the result, efficiency of the membrane will be expressed in terms of the degree of coupling.

Abstrakt. Bude uveden jednoduchý model difúzního typu pro palivový článek s polymerovou membránou (PEMFC). Na základě jeho termodynamické analýzy bude odvozen vztah mezi materiálovými vlastnostmi a geometrickými rozměry membrány. Dále bude analyzována provázanost mezi difúzním tokem a elektrickým proudem v membráně palivového článku. Výsledkem bude vyjádření účinnosti membrány pomocí stupně provázanosti.

1 Introduction

Fuel cells represent a prospective, efficient, and clean alternative to traditional ways of generating electrical energy for automotive, portable and stationary applications [3]. In a so-called proton exchange membrane fuel cell (PEMFC), two electrodes are separated with an electrolyte, which is a thin membrane made of a proton conductive polymer, such as Nafion — a perfluorosulfonic acid manufactured by DuPont. At the anode hydrogen is oxidized, producing protons and electrons. The electrons flow through an external circuit, while the protons must pass through the electrolyte membrane. At the cathode the electrons and the protons react with supplied oxygen to form water. The electrodes are attached to the membrane, forming the so-called membrane electrode assembly (MEA). Transport processes occurring in MEA are considered to be a crucial point of the fuel cell operation and efficiency. The framework of irreversible thermodynamics allows to model all relevant processes like convection, diffusion, heat transfer, ionic conductivity, and electrochemical reactions in a natural way, i. e. based on the entropy production concept. For instance, based on an expression for entropy production density such as

$$\sigma(s) = \boldsymbol{j}_q \cdot \nabla \frac{1}{T} - \sum_\alpha \boldsymbol{j}_{\mathrm{D},\alpha} \cdot \nabla \frac{\mu_\alpha}{T} + \frac{\boldsymbol{i}_\alpha}{F} \cdot \nabla \frac{z_\alpha F \phi}{T} + \frac{\mathbf{t}_{\mathrm{d}is}^{\mathrm{T}}}{T} \cdot \mathbf{D}\boldsymbol{v} + \frac{1}{T} \sum_\rho r_\rho A_\rho,$$
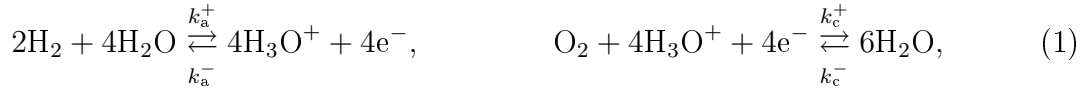
one could derive a very complex model involving diffusion, heat tranfer, migration due to an electrostatic field, friction and chemical reactions. In this paper, we describe a more

simple model that of all these processes takes into account only diffusion and migration, and show on two examples that even a simple thermodynamical analysis can be utilized to produce relevant conclusions.

2 Thermodynamical analysis of a simple diffusion model

2.1 Model equations

Let us introduce a simple diffusion-type model of a hydrogen PEM fuel cell. We consider the cell as a one-dimensional electrochemical reactor into which hydrogen and oxygen are fed and water (and electrical energy) is produced. We consider the electrochemical half-reactions occuring at the electrodes of the cell in the following form [12]:

$$2H_2 + 4H_2O \underset{k_a^-}{\overset{k_a^+}{\rightleftharpoons}} 4H_3O^+ + 4e^-, \qquad O_2 + 4H_3O^+ + 4e^- \underset{k_c^-}{\overset{k_c^+}{\rightleftharpoons}} 6H_2O, \qquad (1)$$

where the former reaction (the oxidation of H_2) is the anode half-reaction, and the latter reaction (the reduction of O_2) is the cathode half-reaction. Mass balances of H_2O and H_3O^+ read (cf., e. g., [13], [14], [15]):

$$\frac{\partial c_{H_2O}}{\partial t} = -4r_a + 6r_c - \operatorname{div} \boldsymbol{j}_{H_2O}, \qquad \frac{\partial c_{H_3O^+}}{\partial t} = 4r_a - 4r_c - \operatorname{div} \boldsymbol{j}_{H_3O^+}, \qquad (2)$$

respectively. Here, c_{H_2O} and $c_{H_3O^+}$ are concentrations of the respective species, $\boldsymbol{j}_{H_2O}$ and $\boldsymbol{j}_{H_3O^+}$ are molar flux densities of the respective species, r_a is the anode reaction rate, and r_c is the cathode reaction rate. For the reaction rates, we have from the mass action law

$$r_a = k_a^+ c_{H_2O}^4 c_{H_2}^2 - k_a^- c_{H_3O^+}^4 c_{e^-}^4, \qquad r_c = k_c^+ c_{O_2} c_{H_3O^+}^4 c_{e^-}^4 - k_c^- c_{H_2O}^6. \qquad (3)$$

For flux densities we have linear transport equations arising from irreversible thermodynamics,

$$\boldsymbol{j}_{H_2O} = -L_{ww} \frac{\partial \mu_\alpha}{\partial c_{H_2O}} \nabla c_{H_2O} - L_{we} \frac{\partial \mu_\alpha}{\partial \phi} \nabla \phi = -L_{ww} \frac{RT}{c_{H_2O}} \nabla c_{H_2O} - L_{we} F \nabla \phi, \qquad (4)$$

$$\boldsymbol{j}_{H_3O^+} = -L_{ew} \frac{\partial \mu_\alpha}{\partial c_{H_2O}} \nabla c_{H_2O} - L_{ee} \frac{\partial \mu_\alpha}{\partial \phi} \nabla \phi = -L_{ew} \frac{RT}{c_{H_2O}} \nabla c_{H_2O} - L_{ee} F \nabla \phi, \qquad (5)$$

where μ_α denotes electrochemical potential, ϕ denotes electrostatic potential, R denotes the universal gas constant, T denotes temperature, F denotes Faraday constant, while L_{ww}, L_{we}, L_{ew} and L_{ee} denote phenomenological transport coefficients. The terms consisting of partial derivatives of μ_α ensure that all the phenomelogical coefficient are of the same physical dimensions $(\text{mol}^2 \text{J}^{-1} \text{m}^{-1} \text{s}^{-1})$. The transport coefficients can be expressed in terms of measurable quantities D_{H_2O} (diffusion coefficient of water), σ (electrical conductivity), and n^d (drag coefficient) as follows:

$$L_{ww} = \frac{D_{H_2O}}{RT} c_{H_2O}, \qquad (6)$$

$$L_{ee} = \frac{\sigma}{F^2}, \qquad (7)$$

$$L_{we} = L_{ee} n^d = \frac{\sigma}{F^2} n^d. \qquad (8)$$

It follows from Onsager reciprocity relations that the "cross" coefficients are equal to each other, i. e.,

$$L_{\mathrm{ew}} = L_{\mathrm{we}}. \tag{9}$$

The relations $(2) - (9)$ together with appropriate functional forms for the transport parameters L_{ww}, L_{we}, L_{ew}, L_{ee} (or $D_{\mathrm{H_2O}}$, σ, n^{d}), and boundary and initial conditions form a simple diffusion-type model of a PEM fuel cell. The main difference between this model and most models that can be found in the literature (cf., e. g., [17], [2] and many later works) is that this model does not follow the usual sandwich structure. In other words, here, the whole fuel cell is described by a single set of equations rather than dividing it into several regions with different properties.

2.2   Membrane characteristic dimension

Now, we shall proceed with linearization of the problem $(2) - (5)$, and perform a qualitative analysis of the linearized problem. We shall follow the method described in [5] as normal mode analysis. To keep things relatively simple, let us make two additional assumptions:

1. The backward reactions in (1) can be neglected, i. e., $k_{\mathrm{a}}^{-} \approx 0$, $k_{\mathrm{c}}^{-} \approx 0$.

2. There is local electroneutrality within the cell, i. e., the rate of oxygen reduction is determined by the concentration of H_3O^+ and O_2.

Under these assumptions, the system $(2) - (5)$ possesses a stationary solution $\left(\overline{c}_{\mathrm{H_2O}}, \overline{c}_{\mathrm{H_3O^+}} \right)$, where

$$\overline{c}_{\mathrm{H_2O}} = \left(\frac{\pi_{\mathrm{w}}}{2k_{\mathrm{a}}^{+} c_{\mathrm{H_2}}^{2}} \right)^{\frac{1}{4}}, \qquad\qquad \overline{c}_{\mathrm{H_3O^+}} = \left(\frac{\pi_{\mathrm{w}}}{2k_{\mathrm{c}}^{+} c_{\mathrm{O_2}}} \right)^{\frac{1}{8}}, \tag{10}$$

$\pi_{\mathrm{w}} = \operatorname{div} \boldsymbol{j}_{\mathrm{H_2O}}$ being production density of water within the cell. Let us recall the charge conservation law,

$$\operatorname{div} \boldsymbol{j} + \frac{\partial \varrho}{\partial t} = 0, \tag{11}$$

where ϱ is charge density. Since in the stationary state $\frac{\partial \varrho}{\partial t} = 0$, we have also

$$\operatorname{div} \boldsymbol{j} = 0. \tag{12}$$

We shall investigate the linearized system $(2) - (5)$ of the following form:

$$\frac{\partial u}{\partial t} = D_{\mathrm{H_2O}} \Delta u - \frac{8\pi_{\mathrm{w}}}{\overline{c}_{\mathrm{H_2O}}} u + \frac{12\pi_{\mathrm{w}}}{\overline{c}_{H_3O^+}} v, \tag{13}$$

$$\frac{\partial v}{\partial t} = D_{\sigma} \Delta u + \frac{8\pi_{\mathrm{w}}}{\overline{c}_{\mathrm{H_2O}}} u + \frac{8\pi_{\mathrm{w}}}{\overline{c}_{H_3O^+}} v. \tag{14}$$

where $D_{\mathrm{H_2O}} = \frac{RT}{\overline{c}_{\mathrm{H_2O}}} L_{\mathrm{ww}}$ is the diffusion coefficient of water, and $D_{\sigma} = \frac{RT}{\overline{c}_{\mathrm{H_2O}}} L_{\mathrm{ew}}$. Let us introduce the following couple of trial functions [5]:

$$u = \overline{c}_{\mathrm{H_2O}} + \delta \mathrm{e}^{ikx+\omega t}, \qquad\qquad v = \overline{c}_{\mathrm{H_3O^+}} + \delta \mathrm{e}^{ikx+\omega t}, \tag{15}$$

where δ, k, and ω are arbitrary complex quantities. By inserting (15) into (13), (14), we obtain

$$\omega u = -D_{\mathrm{w}} k^2 u - \frac{8\pi_{\mathrm{w}}}{\bar{c}_{\mathrm{H_2O}}} u + \frac{12\pi_{\mathrm{w}}}{\bar{c}_{H_3O^+}} v, \tag{16}$$

$$\omega v = -D_{\sigma} k^2 u + \frac{8\pi_{\mathrm{w}}}{\bar{c}_{\mathrm{H_2O}}} u + \frac{8\pi_{\mathrm{w}}}{\bar{c}_{H_3O^+}} v. \tag{17}$$

This system possesses a non-trivial solution, if

$$\begin{vmatrix} -\omega - D_{\mathrm{w}} k^2 - \frac{8\pi_{\mathrm{w}}}{\bar{c}_{\mathrm{H_2O}}} & \frac{12\pi_{\mathrm{w}}}{\bar{c}_{H_3O^+}} \\ -D_{\sigma} k^2 + \frac{8\pi_{\mathrm{w}}}{\bar{c}_{\mathrm{H_2O}}} & -\omega + \frac{8\pi_{\mathrm{w}}}{\bar{c}_{H_3O^+}} \end{vmatrix} = 0, \tag{18}$$

which, in the limit $\omega \to 0$, gives

$$-k^2 = \frac{40\pi_{\mathrm{w}}}{(2D_{\mathrm{w}} - 3D_{\sigma}) c_{\mathrm{H_2O}}}, \tag{19}$$

or,

$$k = \pm \mathrm{i} \sqrt{\frac{40\pi_{\mathrm{w}}}{(2D_{\mathrm{w}} - 3D_{\sigma}) c_{\mathrm{H_2O}}}}. \tag{20}$$

Now, let us equate the imaginary part (or absolute value) of k with the expression $\pm \frac{2\pi}{L}$, where L is a characteristic dimension (thickness) of the membrane. This step has a physical meaning of taking only the first normal mode into account. Such an assumption is reasonable (at least as a first approach) since higher order modes are usually of less significance. Further, let us use an approximation

$$\pi_{\mathrm{w}} = \mathrm{div}\, \boldsymbol{j}_{\mathrm{H_2O}} \approx \frac{2}{L} \frac{i}{F}, \tag{21}$$

where $i = j_{\mathrm{H_3O^+}} F$ is current density; (21) can be justified with the assumption of electroneutrality. Thus, we obtain the formula

$$L \approx \frac{2D_{\mathrm{w}} - 3D_{\sigma}}{20 \boldsymbol{j}_{\mathrm{H_3O^+}}} c_{\mathrm{H_2O}} \pi^2 F. \tag{22}$$

In Fig. 1 are depicted the values of L for a Nafion membrane at the temperature $T = 80\,^{\circ}\mathrm{C}$ and current density $i = 500$ mA/cm^2 calculated using eq. (22) for different values of membrane water content λ, which is the average number of water molecules per SO_3^- group of the Nafion membrane. The values for a highly hydrated membrane around $200\ \mu$m correspond with usual thickness of common fuel cell membranes.

2.3   Degree of coupling

The condition for conversion of energy of one process into energy of another process is a non-zero coupling between these two processes [9]. Coupling is quantitatively described by the degree of coupling. The degree of coupling between diffusion and migration is defined as

$$q = \frac{L_{\mathrm{ew}}}{\sqrt{L_{\mathrm{ww}} L_{\mathrm{ee}}}}. \tag{23}$$
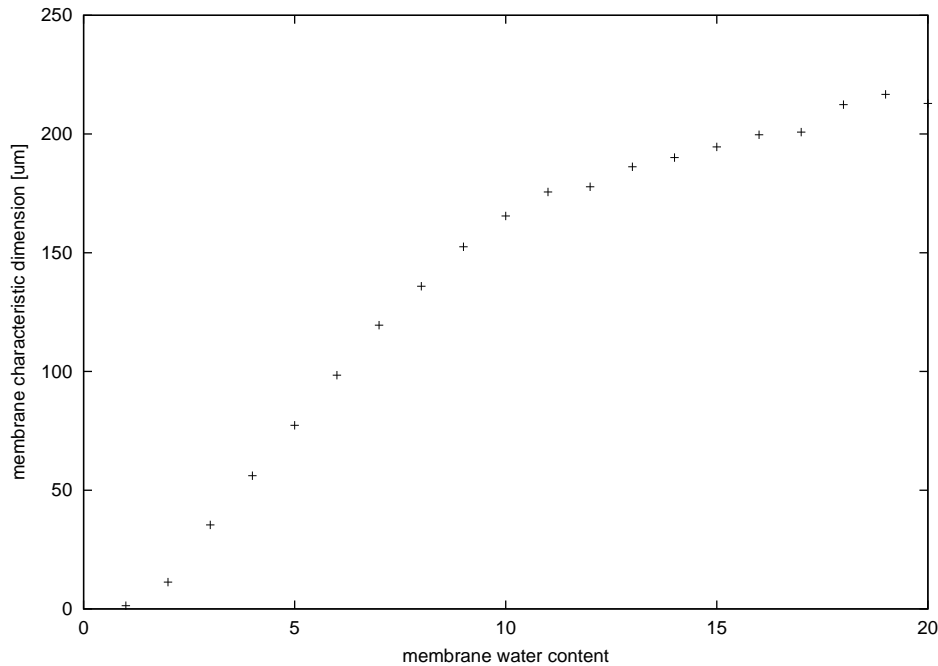
Figure 1: Characteristic dimension (thickness) L of the membrane as a function of the membrane water content λ at the temperature of 80 °C and current density of 500 mA/cm^2. The values of transport coefficients were taken from [4].

The degree of coupling as a function of the membrane water content λ is depicted in Fig. 2. As an application, we shall derive an expression for efficiency of membrane, i. e., efficiency of conversion of chemical energy into electrical energy.

It is well-known that the maximum amount of energy that can be converted into electrical energy during an electrochemical reaction is the Gibbs energy of that reaction. In case of the electrochemical reaction that occurs in fuel cells,

$$H_2 + \frac{1}{2}O_2 \rightarrow H_2O, \tag{24}$$

the molar Gibbs energy is equal to the chemical potential of the water produced during the reaction, since chemical potential of a pure element is by definition equal to zero. Thus, the maximum electrical power per unit area that can be produced from the electrochemical reaction is

$$p_{\text{in}} = -\boldsymbol{j}_{\text{H}_2\text{O}} \cdot \nabla \mu_{\text{H}_2\text{O}}. \tag{25}$$

The actual output power from the fuel cell per unit area is

$$p_{\text{out}} = \boldsymbol{i} \cdot \nabla \phi = z_{\text{H}_3\text{O}^+} F \boldsymbol{j}_{\text{H}_3\text{O}^+} \cdot \nabla \phi, \tag{26}$$

$\boldsymbol{i}$ being the current density, F the Faraday constant, and z_X the charge number of a molecule of species X, which is equal to 1 for H_3O^+ (and also for H^+). Thus, the efficiency of energy conversion can be written down as

$$\eta = -\frac{\boldsymbol{j}_{\text{H}_3\text{O}^+} \cdot F \nabla \phi}{\boldsymbol{j}_{\text{H}_2\text{O}} \cdot \nabla \mu_{\text{H}_2\text{O}}}. \tag{27}$$
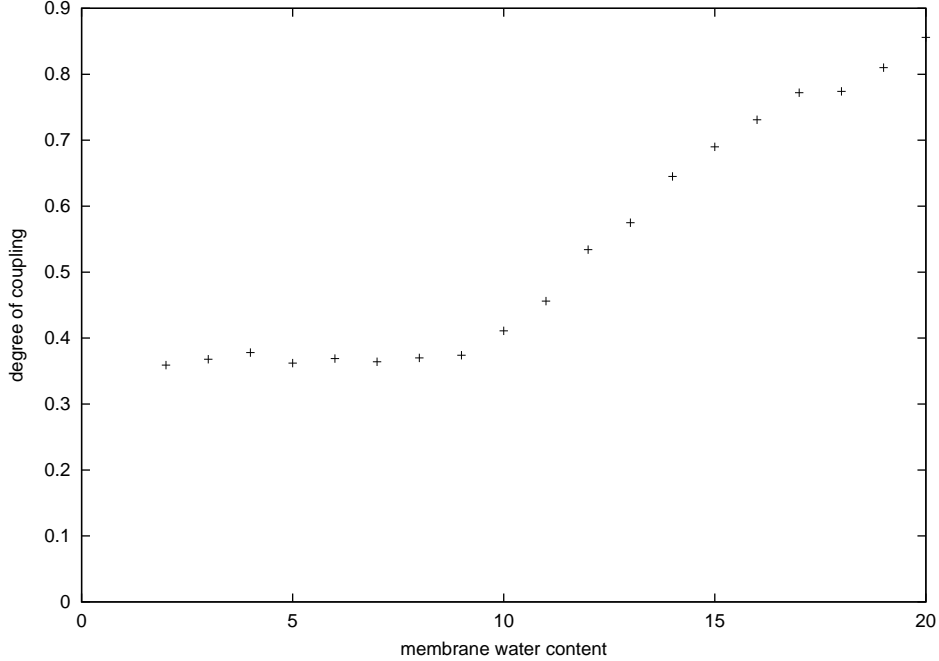
Figure 2: The degree of coupling q as a function of the membrane water content λ at 80 °C. The values of transport coefficients were taken from [4].

From the transport equations (4), (5), we have

$$\eta = -\frac{-L_{ew} - L_{ee}\frac{F\nabla\phi}{\nabla\mu_{H_2O}}}{-L_{ww}\frac{\nabla\mu_{H_2O}}{F\nabla\phi} - L_{we}} = -\frac{L_{ew} + L_{ee}y}{\frac{L_{ww}}{y} + L_{we}} = -\frac{L_{ew}y + L_{ee}y^2}{L_{ww} + L_{we}y}, \qquad (28)$$

where we have denoted $y = \frac{F\nabla\phi}{\nabla\mu_{H_2O}}$. Let us look for the maximum efficiency for any ratio of driving forces. By differentiating (28), we obtain

$$\eta' = -\left(\frac{L_{ew}y + L_{ee}y^2}{L_{ww} + L_{we}y}\right)' = -\frac{(L_{ew} + 2L_{ee}y)(L_{ww} + L_{we}y) - (L_{ew}y + L_{ee}y^2)L_{we}}{(L_{ww} + L_{we}y)^2} =$$

$$= -\frac{L_{ee}L_{we}y^2 + 2L_{ww}L_{ee}y + L_{ww}L_{ew}}{(L_{ww} + L_{we}y)^2}. \quad (29)$$

The nominator in the last expression possesses two real roots, namely

$$-\frac{L_{ww}}{L_{ew}}\left(1 \pm \sqrt{1 - q^2}\right).$$

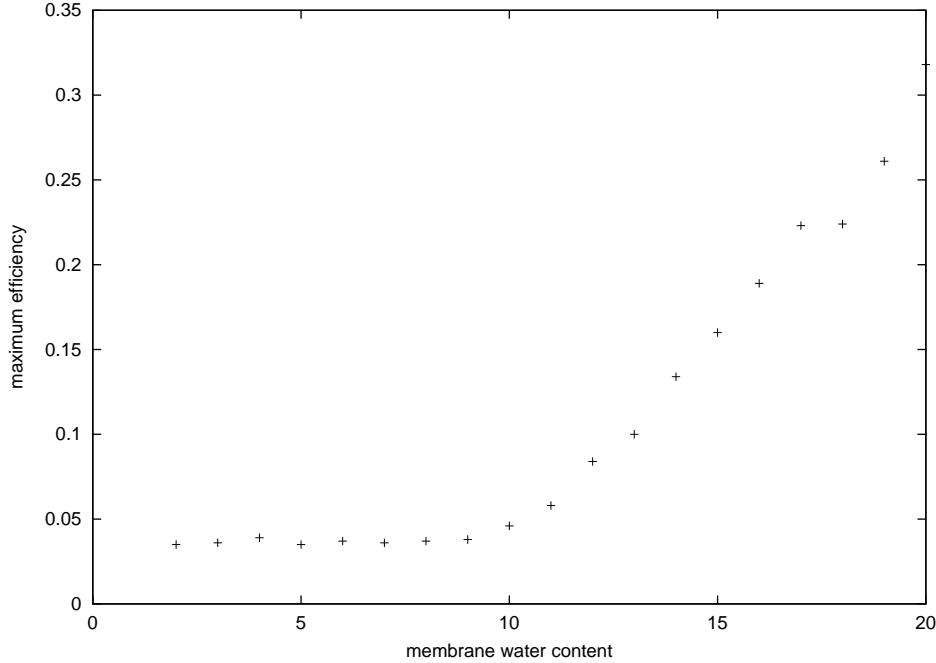Clearly, the more negative one corresponds to a minimum, and the less negative one to a

Figure 3: The maximum efficiency η_{max} as a function of the membrane water content λ at 80 °C. The values of transport coefficients were taken from [4].

maximum. We are interested in the maximum efficiency:

$$
\begin{aligned}
\eta_{\mathrm{max}} &= -\left.\frac{L_{\mathrm{ew}}y + L_{\mathrm{ee}}y^2}{L_{\mathrm{ww}} + L_{\mathrm{we}}y}\right|_{y=-\frac{L_{\mathrm{ww}}}{L_{\mathrm{ew}}}\left(1-\sqrt{1-q^2}\right)} = \\
&= -\frac{-L_{\mathrm{ww}}(1-\sqrt{1-q^2}) + L_{\mathrm{ee}}\frac{L_{\mathrm{ww}}^2}{L_{\mathrm{ew}}^2}(1-\sqrt{1-q^2})^2}{L_{\mathrm{ww}} - L_{\mathrm{ww}}(1-\sqrt{1-q^2})} = \\
&= \frac{1-\sqrt{1-q^2} - \left(\frac{1-\sqrt{1-q^2}}{q}\right)^2}{\sqrt{1-q^2}} = \frac{1-\sqrt{1-q^2}}{\sqrt{1-q^2}}\left(1 - \frac{1-\sqrt{1-q^2}}{q^2}\right) = \\
&= \frac{1-\sqrt{1-q^2}}{\sqrt{1-q^2}}\left(\frac{\sqrt{1-q^2}-(1-q^2)}{q^2}\right) = \left(\frac{1-\sqrt{1-q^2}}{q}\right)^2.
\end{aligned}
$$

We conclude that the maximum efficiency is an increasing function of the absolute value of the degree of coupling. When there is no coupling, the efficiency is zero. In the case of complete coupling, on the other hand, the maximum efficiency tends to 1.

The maximum efficiency as a function of the membrane water content is depicted in Fig. 3. The graph is in accordance with the well-known fact that the membrane has to be well-hydrated in order to obtain reasonable performance (see, e. g., [2], [7], [17], [18]).

2.4 Conclusion

A simple one-dimensional, diffusion-type model of a PEM fuel cell has been presented. Unlike most similar models, this one has attempted to describe both transport processes in the membrane and electrochemical reactions at the electrodes with a single set of equations. An expression for characteristic dimension of the membrane in terms of transport properties has been derived by using linearization and normal mode analysis. The result corresponds with thickness of real fuel cell membranes. Coupling between different transport processes in the Nafion membrane has been analyzed. As the result, an expression for the membrane efficiency has been obtained. The dependence of the efficiency on the membrane water content is in qualitative agreement with empirical findings.

3 Acknowledgments

References

[1] P. W. Atkins, *Physical Chemistry (6th Edition)*. Oxford University Press, New York (1998).

[2] D. M. Bernardi, M. W. Verbrugge, *A Mathematical Model of the Solid-Polymer-Electrolyte Fuel Cell*. J. Electrochem. Soc. **139** (1992), 2477–2491.

[3] E. Birgersson, *Mathematical Modeling of Transport Phenomena in Polymer Electrolyte and Direct Methanol Fuel Cells*. Doctoral Thesis, Royal Institute of Technology, Stockholm (2004).

[4] H. P. L. H. van Bussel, F. G. H. Koene, R. K. A. M. Mallant, *Dynamic model of solid polymer fuel cell water management*. Journal of Power Sources **71** (1998), 218–222.

[5] P. Glansdorff, I. Prigogine, *Thermodynamic Theory of Structure, Stability and Fluctuations*. Wiley – Interscience, London (1971).

[6] H. A. Haus, J. R. Melcher, *Electromagnetic Fields and Energy*. Prentice Hall, Englewood Cliffs (1989).

[7] G. J. M. Janssen, *A Phenomenological Model of Water Transport in a Proton Exchange Membrane Fuel Cell*. J. Electrochem. Soc. **148** (2001), A1313–A1323.

[8] D. Kondepudi, I. Prigogine, *Modern Thermodynamics: From Heat Engines to Dissipative Structures*. Wiley, Chichester (1998).

[9] S. Koter, C. H. Hamann, *Characteristics of Ion-Exchange Membranes for Electrodialysis on the Basis of Irreversible Thermodynamics.* J. Non-Equilib. Thermodyn. **15** (1990), 315–333.

[10] R. Krishna, J. A. Wesselingh, *The Maxwell-Stefan approach to mass transfer.* Chemical Engineering Science **52** (1997), 861–911.

[11] A. A. Kulikovsky, *Quasi-3D Modeling of Water Transport in Polymer Electrolyte Fuel Cells.* J. Electrochem. Soc. **150** (2003), A1432–A1439.

[12] F. Meier, *Membran-Brennstoffzellen Teil II.* Lecture notes for Seminar Membran-Brennstoffzellen-Systeme. University of Stuttgart, Stuttgart (2004).

[13] O. Mičan, *Abstraktní funkce a jejich aplikace ve slabé formulaci smíšené úlohy pro reakčně-difúzní rovnice (rešeršní práce).* Czech Technical University, Prague (2001).

[14] O. Mičan, *Soustavy reakčně-difúzních rovnic a jejich numerické řešení (výzkumný úkol).* Czech Technical University, Prague (2002).

[15] O. Mičan, *Chaotická dynamika reakčně-difúzních rovnic (diplomová práce).* Czech Technical University, Prague (2003).

[16] T. Schultz, *Experimental and Model-based Analysis of the Steady-state and Dynamic Operating Behaviour of the Direct Methanol Fuel Cell (DMFC).* Dissertation Thesis, Otto von Guericke University, Magdeburg (2004).

[17] T. E. Springer, T. A. Zawodzinski, S. Gottesfeld, *Polymer Electrolyte Fuel Cell Model.* J. Electrochem. Soc. **138** (1991), 2334–2342.

[18] A. Z. Weber, J. Newman, *Modeling Transport in Polymer-Electrolyte Fuel Cells.* Chem. Rev. **104** (2004), 4679–4726.

Model of Preferences over the Relational Data Model[*]

Radim Nedbal

4th year of PGS, email: `radned@seznam.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Július Štuller, Institute of Computer Science, Academy of Sciences
of the Czech Republic

Abstract. The aim of the paper is to present a novel, general approach to preference modelling in
the framework of the relational data model. The preferences are defined between sets of relational
instances, which, together with the generalized relational algebra, presents a generalization of
the approach aiming at incorporating partial order into attribute domains and relational algebra.

The main goals are: an intuitive definition of user preferences or a partial order representing
a piece of order related information; an effective representation of a partial order and its efficient
processing throughout the query execution plan; and a suitable data structure design to support
the proposed model.

Abstrakt. Cílem příspěvku je prezentovat nový, obecný přístup k modelování preferencí nad
relačním datovým modelem. Preference jsou definovány na množině instancí relací, což je spolu
s rozšířenou relační algebrou zobecněním přístupu spočívajícího v začlenění uspořádání do domén
atributů a relační algebry.

Hlavním cílem je: intuitivní definice preferencí uživatele a informace reprezentovatelné us-
pořádáním; efektivní reprezentace uspořádání a operací rozšířené relační algebry a návrh vhodné
datové struktury pro navrhovaný model.

1 Related Work

Lacroix and Lavency [8] originated the study of preference queries. They proposed an
extension of the relational calculus in which preferences for tuples satisfying given logical
conditions can be expressed. For instance, one could say: pick up the tuples of the
relation R satisfying the condition $Q \wedge P1 \wedge P2$; if the result is empty, pick the tuples
satisfying the condition $Q \wedge P1 \wedge \neg P2$; if the result is empty, pick the tuples satisfying
the condition $Q \wedge \neg P1 \wedge P2$.
The composition or iteration of preferences, however, is not considered. Neither is ad-
dressed the issue of algebraic optimization of preference queries.

Kießling et al. [6] and Chomicki et al. [4] proposed independently a similar frame-
work based on a formal language for formulating preference relations. The embedding
(called *Best Match Only* – BMO and *WinNow* – WN, respectively) into relational query
languages they use is identical. Many possible rewritings for preference queries are pre-
sented. Kießling et al. [6, 7] introduced a number of base preference constructors and

their combinators (Pareto and lexicographic composition, intersection, disjoint union, and others). Whereas the possibility of having arbitrary constraints in preference formulas is not considered, the framework of Chomicki et al. [4] emphasizes the view of preferences as strict partial orders and defines preferences more generally as arbitrary logical formulas. Intrinsic and extrinsic classes of preference formulas are studied.

Börzsönyi et al. [3] introduced the skyline operator and described several evaluation methods for this operator. Skyline is a special case of WN and BMO.
A declarative query interface for Web repositories that supports complex expressive Web queries is defined in [13]. The Web repository is modelled in terms of "Web relations". A description of an algebra for expressing complex Web queries is given, and an overview of the cost-based optimizer and execution engine is presented.

In [11], actual values of an arbitrary attribute are allowed to be partially ordered to represent a user preference. Accordingly, relational algebra operations, aggregation functions and arithmetic are redefined. Nevertheless, the redefinition of the relational operations causes loss of some of their common properties. For instance, the equality $A \cap B = A - (A - B)$ does not hold. To rectify this weak point, more general approach is needed.

Argawal and Wimmers [1] use quantitative preferences (scoring functions) in queries and focus on the issues arising in combining such preferences. Hristidis et al. [5] explore in this context the problems of efficient query processing using materialized views. As pointed out repeatedly in their paper, the approach based on scoring functions is inherently less expressive than the one based on preference relations. In particular, skyline queries cannot be captured using scoring functions. In addition, it is not clear how to compose scoring functions to achieve an effect similar to various preference relation composition operations.

A different aim is pursued in [12], where the relational data model is extended to incorporate partial orderings into data domains. Within the extended model, the partially ordered relational algebra (the PORA) is defined by allowing the ordering predicate to be used in formulae of the selection operation. The expressiveness of the PORA is formally shown and the development of Ordered SQL (OSQL) as a query language for ordered databases is justified.

A comprehensive work on partial order in databases is [15]. It presents the partially ordered sets as the basic construct for modelling data. Collection of algebraic operations for manipulating ordered sets is investigated, and their implementation based on the use of realizers as a data structure is presented. An algorithm for generating realizers for arbitrary partial orders is provided.

In the context of financial and statistical applications, systems such as SEQUIN [16], SRQL [14], and more recently Aquery [10, 9] have proposed SQL extensions to incorporate ordering. They aim at exploiting linear order inherent in many kind of data, i.e., time series., in query processing.

2 Introduction

2.1 The Relational Data Model

The term *relational data model* has come to refer to a broad class of database models that have relations as the data structure and that incorporate query capabilities, update capabilities, and integrity constraints. Intuitively, the data is represented in tables in which each row contains data about a specific object or a set of objects, and rows with uniform structure and intended meaning are grouped into tables. Updates consist of transformations of tables by insertion, deletion, or modification of rows. Queries allow the extraction of information from tables. A fundamental feature of virtually all relational query languages is that the result of a query is also a table.

Each table, representing a relation, has a name. Furthermore, the corresponding table columns are called attributes and also have names. Each line in a table represents a tuple. The entries of tuples are taken from sets – domains of corresponding attributes, e.g., $Dom(A)$ is the domain of attribute A. Finally, we distinguish between the database scheme, which specifies the structure of the database; and the database instance, which specifies its actual content.

More formally, a *database schema* is a nonempty finite set of *relation schemas*. A relation schema, denoted $R[U]$, is a pair of relation name R and a set U of attributes. Tuples can be viewed as functions (under the so-called *named perspective* of relational model). More precisely, a tuple t over a relation schema $R[U]$ is a mapping from U to the union of domains of attributes of U such that $\forall A \in U(t(A) \in Dom(A))$. Finally, a relational instance over a relational schema $R[U]$ is (under the so-called *conventional perspective*) a finite set R^* of tuples over $R[U]$.

2.2 Notation

R, S, T	relation names
R^*, S^*, T^*	relation instances
$R[U]$	relation schema with set U of attributes
r, s, t	tuples
$(R, \sqsubseteq_R)$	partial order on the set of instances of relation R
P	preference
$P(R^*)$	preference P over subsets of R^*

2.3 The Main Goals

Let us start with the following illustrative and motivating example.

Example 1. Consider a query whose output should be a group of people working for a university "Uni-S". A requirement is no superior – subordinate relationship between any two members of the group. Furthermore, the preference for superior employees in the superior – subordinate hierarchy should be taken into account.

- The superior – subordinate hierarchy, depicted in Fig.1, presents a piece of information we have about the domain. It is representable by means of a partial order

over the positions at the university.
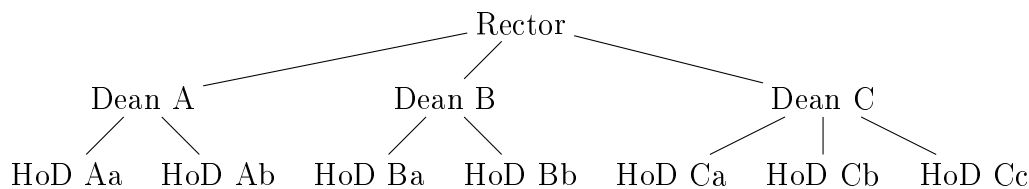
Explanation:
     Dean A, Dean B, Dean C   – dean of A, dean of B, dean of C
     HoD Aa, HoD Ab, . . . , HoD Cc – head of department Aa, Ab, . . . , Cc

Figure 1: superior – subordinate hierarchy of a university

- The requirement for no superior – subordinate relationship between any two members of the group can be represented[1] as a preference P_1 on the powerset of employees of the university in question.

- The preference for superior employees in the superior – subordinate hierarchy can be expressed by means of another preference P_2.

What is the relationship between the above preferences P_1 and P_2? Observe that their importance is unequal from the viewpoint of our query. The requirement for no superior – subordinate relationship presents the higher priority preference. Thus the overall preference will be expressed by means of a preference combinator & reflecting this relationship[2]:
$$P = P_1 \ \& \ P_2$$

    □

In general, a user preference can become quite complex. Also, the order representing a piece of information might be much more complicated then the one considered in the above example. To sum up, the following problems arise:

- How can be defined a partial order representing a piece of order related information or user preferences intuitively?

- How can be represented and processed throughout the query execution plan a partial order information effectively?

- What data structure is suitable to support the proposed model?

3   The Proposed Solution

The mathematical theory deals with partial orders over infinite sets, but we restrict ourself to partial orders over finite sets in the same way that the relational data model restricts itself to finite relations (in particular, this means that the maxima and minima are always

---

[1]For more detailed discussion on constructing preferences refer to Subsection 3.2.
[2]For more detailed discussion on combining preferences refer to Subsection 3.2.

well-defined). Furthermore, we respect a closed world assumption with respect to order representing a user preference, which means that elements of the world not belonging to the base set of the partial order are assumed to be minimal.

3.1 Preference Modelling

Broadly speaking, a preference of a state can be understood as its proximity to the optimal state. A state can be described by its model. Similarly, a derived state can be described by the model of the state it is derived from and by a set of deriving logical formulas. As the model is a subset of a Herband base, it can be viewed as a theory. If this theory in conjunction with a set of deriving formulas has a minimal model that describes the derived state in question, then the proximity of a state derived from a suboptimal one to a state derived from the optimal one by the same set of formulas equates to the proximity of the suboptimal state to the optimal one. The higher proximity can be assigned to a state, the higher preference of the state can be concluded.

Note that, generally, a state might be a derivative from n other states with various preferences. Then the derived state is assigned $n-$tuple of proximities of the corresponding states.[3]

According to the above concept, a relation corresponds to a state, a relational instance to its model, and a query expression to a deriving logical formula. For example, a selection condition can be used to derive a state that we are interested in.

3.2 Preference Engineering

The goal is to provide intuitive and convenient ways to inductively construct a partial order $P = (R, \sqsubseteq_R)$ representing a desired preference. We distinguish between base and compound preferences. The inductive construction is based on the notion of *preference term*.

Definition 2 (Preference term). Given preference terms P_1 and P_2, P is a *preference term* iff P is one of the following:

- Base preference.

- Subset preference: $P := P_1^{\subseteq}$., i.e. a restriction of P_1.

- Dual preference: $P := P_1^{\partial}$., i.e., defined by a dual partial order.

- Complex preference gained by applying one of the following preference combinators:

  - Pareto accumulation: $P := P_1 \otimes P_2$.
  - Prioritized accumulation: $P := P_1 \& P_2$.

---

[3]Thus in case of a lattice ordering of proximities, the preference of a state can be determined by the least upper bound of all the proximities that can be assigned to the state.

Base Preferences. The set of Base preference constructors is extensible, if required by the application domain. Commonly useful constructors include the following:

- POS(SET) specifies that a given SET of values should be preferred.

- NEG(SET) specifies that a given SET of values should be avoided.

- $POS \setminus POS(SET_1, SET_2)$ specifies two sets of values that should by preferred. At the same time, values of the first set SET_1 should be preferred also to values of the second set SET_2.

- $POS \setminus NEG(SET_1, SET_2)$ specifies two sets of values. The values from the first set SET_1 should be preferred, and the values from the second set SET_2 should be avoided.

- EXPLICIT is represented explicitly, e.g., by means of a realizer.

- PF, preference formula, is a first order formula defining a preference relation in the standard sense, namely

$$S^* \sqsupseteq_{PF} T^* \equiv PF(S^*, T^*) \ .$$

Complex Preferences. It is possible to construct more complex preferences by means of *preference combinators*. They can combine preferences coming from one or several parties.

Definition 3 (Pareto preference: $P_1 \otimes P_2$). Both preferences P_1 and P_2 are considered to be equally important:

$$R^* \sqsubseteq_{P_1 \otimes P_2} S^* \equiv R^* \sqsubseteq_{P_1} S^* \wedge R^* \sqsubseteq_{P_2} S^* \ .$$

Definition 4 (Prioritized preference: $P_1 \& P_2$). P_1 is considered more important than P_2, which is respected only when P_1 does not mind:

$$R^* \sqsubseteq_{P_1 \& P_2} S^* \equiv R^* \sqsubseteq_{P_1} S^* \vee (R^* = S^* \wedge R^* \sqsubseteq_{P_2} S^*) \ .$$

Example 5. Recall the Example 1. We defined the preference P by means of prioritized preference combinator:

$$P = P_1 \ \& \ P_2 \ ,$$

where P_1 and P_2 were the base preferences. They can be defined as:

- a NEG-preference $P_1 := \text{NEG}(\text{NEG-set})$

- and a PF preference $P_2 := PF(S^*, T)$, respectively,

where the NEG-set is defined by means of a logical formula:

$$\text{NEG-set} = \{R^{*\prime} \subseteq R^* | \exists r_1, r_2 \in R^{*\prime}(r_1 \sqsubseteq_R r_2)\} \ ,$$

and preference formula PF is defined by means of preference logical formula:

$$PF(S^*, T^*) = \forall t \in T^*(t \in S^* \vee \exists s \in S^*(t \sqsubseteq_R s)) \ .$$

Note that both logical formulas contain a predicate symbol $\sqsubseteq_R$ interpreted by means of partial order representing a piece of knowledge we have about the knowledge domain. □

3.3   Partial Order Model

The base set elements of a partial order can be arbitrary objects. In the proposed model, the objects are sets, namely the relational instances. The aim is to model order on a set of relational instances, which is a nontrivial generalization of the approach aiming at incorporating partial ordering into attribute domains and the corresponding extended relational algebra.

Partial Order Algebra.   The partial orders are to be manipulated as objects in an algebra that should be closed. That is, the result of applying any operation to any partial order should be a new partial order. At least, the *partial order algebra* should include:

- **Predicates** determining satisfaction of a given property, among others: (binary) **containment** and **equality**.

- **Operators**, among others: **selection**, **duplicate elimination**, **maxima** and **minima**, extracting the maximal and minimal elements respectively, **up-tail** and **down-tail**, extracting the suborder that result by removing the minima and maxima respectively.

- **Operations**, among others (unary) **projection** and (binary) **cartesian product**, **intersection**, **union**, **difference**.

Partial Order Implementation.   The implementation of the partial order model includes both a data structure to hold the partial orders, and algorithms for the algebra operations. A partial order can be represented by means of *realizer*, which is shown to simplify the algorithms.

Definition 6 (Realizer). Realizer is a set of linear extensions of the partial order such that for any incomparable elements $a \parallel b$ there exist a linear extension $<_i$ in which $a <_i b$ and a linear extension $<_j$ in which $b <_j a$.

Example 7 (Realizer). Figure 2 shows a partial order and its realizer, consisting of two linear extensions.   □
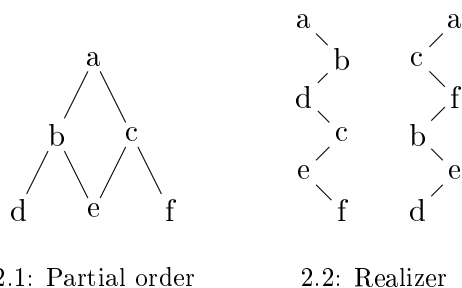

2.1: Partial order   2.2: Realizer

Figure 2: Partial order and its realizer

Remark 8. A partial order realizer is a collection of sequences, each of which contains all the elements of the partial order base set. For this reason, an array is a suitable data structure to hold the realizer.

3.4   Preference Model Implementation

The aim of this subsection is to present the data structure to effectively implement the proposed partial order model. As the semantics of partial order defining preferences is vertically oriented, a data structure with a column oriented semantics in which variables are bound to arrays, not tuples, should be employed.

A promising solution is *arrable*, for array-table, which is an ordered data structure. Informally, it is a collection of named arrays whose values may be arrays themselves. Essentially, arrable is a table organized by columns.

Definition 9 (Arrable). Let $\mathscr{T}$ be a set of types corresponding to a basic type or to a one-dimensional array of basic type elements. Let A be a finite array of elements of a type from $\mathscr{T}$. The cardinality of A is the number of elements in A's first dimension. $A[k]$ is the $k-$th element, and k is said to be an *index* or *position* in A. An *arrable* is a collection of named arrays $A_1, \ldots, A_n$, each being of a type from $\mathscr{T}$, that have the same cardinality.

Each relational algebra operator takes array-typed expressions as arguments and an according execution model should be provided.

The most common execution model is called *iterator-based*. It is however cache inefficient due to loading unnecessary columns' data to perform an operation. For instance, even if the projection involves only one attribute, it brings the entire tuple to occupy cache space.

A contrasting execution model is the *column-oriented* one found in [2]. The essence of the model consists in that instead of handling a row at a time, data is vertically partitioned in columns that are manipulated as units. For instance, selection looks at the columns involved in the selection condition and returns the IDs of rows satisfying the condition. It is not necessary to materialize result of each operation. Instead, the reference to the resulting data can be passed on to the next operation.

4   Conclusion

4.1   Summary and Results

A generalized model of preferences for the relational data model has been proposed. It should present a basis for semantically rich, easy to handle and flexible preference model aiming at deep personalization of database queries.

- The biggest difference, as compared with other mentioned approaches, consists in **expressing preferences by means of a partial order on a set of relational instances**, while the other known approaches express preferences by means of a partial order on a set of relation tuples or on attribute domains. As a relation tuple is a special case of a corresponding relation instance, the presented approach is strictly more general.

- Another key difference is a **use of realizers to encode preferences**. Realizers present a means to process preference-related information throughout the whole query execution plan. Therefore **no preference-related information is lost** as compared to

the known approaches aiming at only the best preference-matching results. Thus a bigger expressivity is reached.

- Realizers are employed also to effectively hold all kinds of information expressible by partial order, and corresponding algebra is proposed to manipulate them. This is the prerequisite for allowing **a selection condition to contain predicates interpreted by partial orders**. It has been shown [12] that **this increases the expressivity** of the resulting *partial order relational algebra*.

- Also, **a data structure, called arrable** and first introduced in [2], is proposed to implement the proposed preference model. It is a column oriented data structure, **suitable for data representing information with "column-oriented semantics"**, the semantics considered in the proposed preference model. Moreover, queries over arrables are executed by breaking them down into a sequence of array primitives, which are often small enough to fit into the main memory. It has a considerable implications concerning **efficient query processing**.

4.2   Future Work

It can be shown that, in the framework of modelling preferences by means of partial order on relation instances, the *partially ordered relational algebra* operations can be well-defined in the sense that all the known identities hold. Specifically, associativity and commutativity of the union, product, selection, and projection operators are retained. Moreover, the following equivalences hold:

$$
\begin{aligned}
\sigma_{\varphi_1 \vee \varphi_2}(\mathcal{R}) &\equiv \sigma_{\varphi_1}(\mathcal{R}) \cup \sigma_{\varphi_2}(\mathcal{R}) \ , \\
\sigma_{\varphi_1 \wedge \varphi_2}(\mathcal{R}) &\equiv \sigma_{\varphi_1}(\mathcal{R}) \cap \sigma_{\varphi_2}(\mathcal{R}) \ , \\
\sigma_{\neg \varphi}(\mathcal{R}) &\equiv \mathcal{R} - \sigma_{\varphi}(\mathcal{R}) \ .
\end{aligned}
$$

Using the proposed approach, other relational operators (intersection, join, and division), also, retain the usual properties of their classical relational counterparts:

$$
\begin{aligned}
\mathcal{R} \cap \mathcal{S} &\equiv \mathcal{R} - (\mathcal{R} - \mathcal{S}) \ , \\
\mathcal{R} \div \mathcal{S} &\equiv \pi_{A-B}(\mathcal{R}) - \pi_{A-B}\Big(\big(\pi_{A-B}(\mathcal{R}) \times \mathcal{S}\big) - \mathcal{R} \Big) \ , \\
\mathcal{R} \bowtie \mathcal{S} &\equiv \pi_{A \cup B}\big(\sigma_{\varphi}(\mathcal{R} \times \mathcal{S}) \big) \ .
\end{aligned}
$$

These results are promising towards the **query optimization issues**, which present many open problems. In particular,

- evaluation and optimization of preference queries, including cost-base optimization, need to be addressed.

- Besides, there is an open problem of exploiting the combination of the iterator-based and the column-oriented execution model. The crux probably consists in stepwise realizer construction, modelling stepwise adding pieces of preference-related information, and the related impact against the partially ordered relational algebra operations.

References

[1] R. Agrawal and E. Wimmers. A Framework for Expressing and Combining Preferences. In 'SIGMOD Conference', W. Chen, J. F. Naughton, and P. A. Bernstein, (eds.), 297–306. ACM, (2000).

[2] P. A. Boncz. *Monet: A Next-Generation DBMS Kernel For Query-Intensive Applications.* Ph.d. thesis, Universiteit van Amsterdam, Amsterdam, The Netherlands, (May 2002).

[3] S. Börzsönyi, D. Kossmann, and K. Stocker. The skyline operator. In 'Proceedings of the 17th International Conference on Data Engineering', 421–430, Washington, DC, USA, (2001). IEEE Computer Society.

[4] J. Chomicki. *Preference Formulas in Relational Queries.* ACM Trans. Database Syst. **28** (2003), 427–466.

[5] V. Hristidis, N. Koudas, and Y. Papakonstantinou. Prefer: a system for the efficient execution of multi-parametric ranked queries. In 'SIGMOD '01: Proceedings of the 2001 ACM SIGMOD international conference on Management of data', 259–270, New York, NY, USA, (2001). ACM Press.

[6] W. Kießling. Foundations of Preferences in Database Systems. In 'Proceedings of the 28th VLDB Conference', 311–322, Hong Kong, China, (2002).

[7] W. Kießling. Preference constructors for deeply personalized database queries. Technical Report 2004-07, Institute of Computer Science, University of Augsburg, (March 2004).

[8] M. Lacroix and P. Lavency. Preferences; Putting More Knowledge into Queries. In 'VLDB', P. M. Stocker, W. Kent, and P. Hammersley, (eds.), 217–225. Morgan Kaufmann, (1987).

[9] A. Lerner. *Querying Ordered Databases with AQuery.* PhD thesis, ENST-Paris, France, (2003).

[10] A. Lerner and D. Shasha. Aquery: Query language for ordered data, optimization techniques, and experiments. In '29th International Conference on Very Large Data Bases (VLDB'03)', 345–356, Berlin, Germany, (September 2003). Morgan Kaufmann Publishers.

[11] R. Nedbal. *Relational Databases with Ordered Relations.* Logic Journal of the IGPL **13** (2005), 587–597.

[12] W. Ng. *An Extension of the Relational Data Model to Incorporate Ordered Domains.* ACM Transactions on Database Systems **26** (September 2001), 344–383.

[13] S. Raghavan and H. Garcia-Molina. Complex queries over web repositories, (2003). `citeseer.ifi.unizh.ch/raghavan03complex.html`.

[14] R. Ramakrishnan, D. Donjerkovic, A. Ranganathan, K. S. Beyer, and M. Krishnaprasad. Srql: Sorted relational query language. In 'SSDBM '98: Proceedings of the 10th International Conference on Scientific and Statistical Database Management', 84–95, Washington, DC, USA, (1998). IEEE Computer Society.

[15] D. R. Raymond. *Partial-order databases*. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada, (1996). Adviser-W. M. Tompa.

[16] P. Seshadri, M. Livny, and R. Ramakrishnan. The design and implementation of a sequence database system. In 'VLDB '96: Proceedings of the 22th International Conference on Very Large Data Bases', 99–110, San Francisco, CA, USA, (1996). Morgan Kaufmann Publishers Inc.

2D Image Recognition in Frequency Domain

Kateřina Nováková

4th year of PGS, email: `katka.novakova@dc.fjfi.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Jaromír Kukal, Department of Software Engineering in Economy,
Faculty of Nuclear Science and Physical Engineering, CTU

Abstract. Two dimensional objects can be represented as 2D binary images. The paper is oriented to affine invariant recognition of them via 2D Fourier spectrum. The amplitude spectrum of binary image is translation invariant. The moments of second order were used to obtain affine invariant spectrum. The square of 2D spectrum was analyzed on circular paths with radius omega. Harmonic analysis of samples on the path enabled to study even harmonics. The absolute values of Fourier coefficients were studied for $n < 21$ and $e^{-\frac{1}{2}} \leq \omega \leq e^2$. It is easy to prove that the system of coefficients is affine invariant and able to recognize large objects. All the calculations were performed in the Matlab environment. 2D binary objects were studied both theoretically (square, triangle, circle, hexagon) and experimentally. Proposed method was tested on various object classes with various affine transforms. The effect of discretization was also studied. The object recognition can be based on coefficient similarity and PCA technique eliminates redundant dimensionality. The methodology is useful for object recognition in biomedicine and industry.

Abstrakt. Dvoudimenzionální objekty mohou být reprezentovány jako 2D binární obrazy. Práce je orientována na jejich rozpoznávání invariantní vůči afinní transformaci a využívá 2D Fourierovu transformaci. Amplitudové spektrum je invariantní vůči posunutí. Pro dosažení afinně invariantního spektra byly použity momenty druhého řádu. Druhá mocnina tohoto spektra byla analyzována na kružnicích s poloměrem omega. Harmonická analýza signálu na kružnicích umožňuje studovat sudé harmonické kmity. Byly studovány absolutní hodnoty fourierových koeficientů. Systém koeficientů je afinně invariantní a umožňuje rozpoznávání větších objektů. 2D objekty byly studovány teoreticky i experimentálně. Navržená metoda byla testována na různých třídách objektů s různými afinními transformacemi. Současně byl zkoumán vliv diskretizace. Rozpoznávání může být založeno na podobnosti koeficientů a PCA pomáhá při eliminaci nadbytečné dimenze. Metodika je užitečná pro rozpoznávání objektů v biomedicíně a průmyslu.

1 Introduction

Let $N \times N$ be 2D binary image. It can be represented as mapping $f : [0, N]^2 \rightarrow \{0, 1\}$. Then continuous 2D Fourier transform is defined as

$$F(\omega_1, \omega_2) = \int_0^N \int_0^N f(x_1, x_2) e^{-i\omega_1 x_1 - i\omega_2 x_2} \, dx_1 \, dx_2, \tag{1}$$

where $\omega_1, \omega_2 \in \mathbb{R}$, $i^2 = -1$. Now, we define normalized 2D power spectrum as

$$\Phi(\omega_1, \omega_2) = \left| \frac{F(\omega_1, \omega_2)}{F(0, 0)} \right|^2 \tag{2}$$

for non-empty image. The function Φ is invariant to any translation of 2D image. It can be easily proven from the basic properties of Fourier transform. The main question, which is subject of this paper, is about realization of new affine invariant recognition system for 2D binary images. The system will be developed for continuous case first. The approximation for discrete images will be also defined and studied to be useful in real image recognition.

2 Affine invariant system

Let $p, q \in \mathbb{N}_0$ be orders. Then image moments are defined as

$$m_{p,q} = \int_0^N \int_0^N x_1^p x_2^q f(x_1, x_2) \, dx_1 \, dx_2. \tag{3}$$

It is also useful to define three central moments of 2nd order by formulas

$$\mu_{2,0} = \frac{m_{2,0}}{m_{0,0}} - \left(\frac{m_{1,0}}{m_{0,0}} \right)^2, \tag{4}$$

$$\mu_{1,1} = \frac{m_{1,1}}{m_{0,0}} - \frac{m_{1,0}}{m_{0,0}} \frac{m_{0,1}}{m_{0,0}}, \tag{5}$$

$$\mu_{0,2} = \frac{m_{0,2}}{m_{0,0}} - \left(\frac{m_{0,1}}{m_{0,0}} \right)^2. \tag{6}$$

They are defined only for non-empty image when $m_{0,0} > 0$ and the matrix

$$\mathsf{G} = \left(\begin{array}{cc} \mu_{2,0} & \mu_{1,1} \\ \mu_{1,1} & \mu_{0,2} \end{array} \right). \tag{7}$$

is positive definite with eigenvalues $\lambda_1 \geq \lambda_2 > 0$ and eigenvectors $\overline{e}_1, \overline{e}_2$.

Now, the second order approximation of 2D power spectrum is

$$\Phi(\omega_1, \omega_2) \approx 1 - \mu_{2,0}\omega_1^2 - 2\mu_{1,1}\omega_1\omega_2 - \mu_{0,2}\omega_2^2. \tag{8}$$

The spectrum is radial symmetric just when $\mu_{2,0} = \mu_{0,2}$ and $\mu_{1,1} = 0$. To guarantee the radial symmetry of new spectrum, we use eigenvalue decomposition (EVD) of matrix G and calculate transformation matrix

$$\mathsf{A} = \mathsf{Q}\mathsf{L}^{-\frac{1}{2}}, \tag{9}$$

where

$$\mathsf{Q} = \left(\overline{e}_1 | \overline{e}_2 \right), \tag{10}$$

$$\mathsf{L} = \left(\begin{array}{cc} \lambda_1 & 0 \\ 0 & \lambda_2 \end{array} \right). \tag{11}$$

It is easy to construct radial symmetric 2D spectrum

$$\Psi(\omega_1, \omega_2) = \Phi(a_{1,1}\omega_1 + a_{1,2}\omega_2, a_{2,1}\omega_1 + a_{2,2}\omega_2) \tag{12}$$

from the power spectrum and matrix A.

The second order approximation is then

$$\Psi(\omega_1, \omega_2) \approx 1 - \omega_1^2 - \omega_2^2. \tag{13}$$

The 2D spectrum Ψ is both translation and scaling invariant. It is also well prepared for the construction of affine invariant system. We can substitute

$$\omega_1 = \omega \cos \varphi, \tag{14}$$
$$\omega_2 = \omega \sin \varphi \tag{15}$$

for $\omega \geq 0$ and $\varphi \in [0, 2\pi]$. The function $\Psi(\omega \cos \varphi, \omega \sin \varphi)$ is periodic in φ for any fixed ω. Thus, the function can be expressed as Fourier series for given ω. The squared absolute values of Fourier coefficients are

$$C_n(\omega) = \left| \frac{1}{2\pi} \int_0^{2\pi} \Psi(\omega \cos \varphi, \omega \sin \varphi) e^{in\varphi} \, d\varphi \right|^2 \tag{16}$$

for $n \in \mathbb{N}_0$, $\omega \geq 0$. The functions $C_n(\omega)$ are affine invariant with respect to changes in 2D image f. The property $a = \left(\frac{\lambda_2}{\lambda_1} \right)^{\frac{1}{2}} \in (0, 1]$ can be called inverse excentricity. It also enables to recognize 2D binary objects, but the system consisting of inverse excentricity a and functions $C_n(\omega)$ is not affine invariant.

3 Discrete case

The 2D discrete binary image can be represented as mapping $f^+ : \{0, \ldots, N - 1\}^2 \to \{0, 1\}$. The relationship between function f and its discrete form f^+ is defined here as $f(x_1, x_2) \approx f^+(k_1, k_2)$ for $k_1 \leq x_1 < k_1 + 1$ and $k_2 \leq x_2 < k_2 + 1$. So, f^+ is an piecewise constant approximation of 2D image function f. The adequate discrete spectrum is defined as

$$F^+(\omega_1, \omega_2) = \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} f^+(k_1, k_2) e^{-i\omega_1 k_1 - i\omega_2 k_2} \tag{17}$$

and the adequate discrete moments are

$$m_{p,q}^+ = \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} k_1^p k_2^q f^+(k_1, k_2), \tag{18}$$

$$\mu_{2,0}^+ = \frac{m_{2,0}^+}{m_{0,0}^+} - \left(\frac{m_{1,0}^+}{m_{0,0}^+} \right)^2, \tag{19}$$

$$\mu_{1,1}^+ = \frac{m_{1,1}^+}{m_{0,0}^+} - \frac{m_{1,0}^+}{m_{0,0}^+} \frac{m_{0,1}^+}{m_{0,0}^+}, \tag{20}$$

$$\mu_{0,2}^+ = \frac{m_{0,2}^+}{m_{0,0}^+} - \left(\frac{m_{0,1}^+}{m_{0,0}^+} \right)^2. \tag{21}$$

There is also direct relationship between spectrum $F(\omega_1, \omega_2)$ and its discrete analogy $F^+(\omega_1, \omega_2)$. We obtained

$$|F(\omega_1, \omega_2)| \approx \left| \frac{\sin \frac{\omega_1}{2}}{\frac{\omega_1}{2}} \right| \left| \frac{\sin \frac{\omega_2}{2}}{\frac{\omega_2}{2}} \right| |F^+(\omega_1, \omega_2)| \tag{22}$$

This approximation enables to make another Taylor expansion of 2nd order

$$\Phi(\omega_1, \omega_2) \approx 1 - \left(\mu_{2,0}^+ + \frac{1}{12} \right) \omega_1^2 - 2\mu_{1,1}^+ \omega_1 \omega_2 - \left(\mu_{0,2}^+ + \frac{1}{12} \right) \omega_2^2. \tag{23}$$

When the discrete image is non-empty, then $m_{0,0}^+ > 0$ and the matrix

$$\mathsf{G}^+ = \begin{pmatrix} \mu_{2,0}^+ + \frac{1}{12} & \mu_{1,1}^+ \\ \mu_{1,1}^+ & \mu_{0,2}^+ + \frac{1}{12} \end{pmatrix} \tag{24}$$

is also positive definite with eigenvalues $\lambda_1^+ \geq \lambda_2^+ > 0$, eigenvectors $\overline{e}_1^+$, $\overline{e}_2^+$ and inverse excentricity a^+. In analogy with non-discrete case we obtained

$$\mathsf{Q}^+ = \left(\overline{e}_1^+ | \overline{e}_2^+ \right), \tag{25}$$

$$\mathsf{L}^+ = \begin{pmatrix} \lambda_1^+ & 0 \\ 0 & \lambda_2^+ \end{pmatrix}, \tag{26}$$

$$\mathsf{A}^+ = \mathsf{Q}^+ \left(\mathsf{L}^+ \right)^{-\frac{1}{2}}, \tag{27}$$

$$\Phi^+(\omega_1, \omega_2) = \left(\frac{\sin \frac{\omega_1}{2}}{\frac{\omega_1}{2}} \frac{\sin \frac{\omega_2}{2}}{\frac{\omega_2}{2}} \right)^2 \left| \frac{F^+(\omega_1, \omega_2)}{F^+(0,0)} \right|^2, \tag{28}$$

$$\Psi^+(\omega_1, \omega_2) = \Phi^+(a_{1,1}^+ \omega_1 + a_{1,2}^+ \omega_2, a_{2,1}^+ \omega_1 + a_{2,2}^+ \omega_2). \tag{29}$$

Finally, we obtained discrete approximations

$$C_n^+(\omega) = \left| \frac{1}{2\pi} \int_0^{2\pi} \Psi^+(\omega \cos \varphi, \omega \sin \varphi) e^{in\varphi} \, d\varphi \right|^2 \tag{30}$$

for $n \in \mathbb{N}_0$, $\omega \geq 0$. It is clear, that $C_n(\omega) \approx C_n^+(\omega)$ but $C_n^+(\omega)$ are not affine invariant functions. They are only good approximations of exact affine invariant functions $C_n(\omega)$.

4   Implementation details

There are some constrains, which are useful for good approximation power of $C_n^+(\omega)$. The discrete 2D image must be large. It means, that $N \geq 64$ and there is a pixel pair with a distance $d \geq 50$. The functions $C_0(\omega)$, $C_0^+(\omega) \in [0,1]$, $C_0(0) = C_0^+(0) = 1$ and $\lim_{\omega \to \infty} C_0(\omega) = \lim_{\omega \to \infty} C_0^+(\omega) = 0$. The odd harmonic frequences are missing. The

$$\omega \in \mathbb{R}_0^+ \Rightarrow C_{2k+1}(\omega) = C_{2k+1}^+(\omega) = 0. \tag{31}$$

So, it is to nothing to evaluate them. The other even functions $C_{2k}(\omega), C_{2k}^+(\omega)$ for $k \in \mathbb{N}$ satisfy $C_{2k}(0) = C_{2k}^+(0) = \lim\limits_{\omega \to \infty} C_{2k}(\omega) = \lim\limits_{\omega \to \infty} C_{2k}^+(\omega) = 0$. So, the invariant functions $C_{2k}(\omega)$ and their approximations $C_{2k}^+(\omega)$ are useful for object recognition in the range $\omega \in [e^{-\frac{1}{2}}, e^2]$. The relative differences for lower and higher frequency ω are small. The functions can be sampled with frequency ratio $q = e^{0.01}$ to obtain affine invariant vector description

$$\overline{\varphi}_k = (C_{2k}(\omega_{min}), C_{2k}(\omega_{min} q), \dots, C_{2k}(\omega_{max})) \tag{32}$$

or its discrete approximation

$$\overline{\varphi}_k^+ = (C_{2k}^+(\omega_{min}), C_{2k}^+(\omega_{min} q), \dots, C_{2k}^+(\omega_{max})). \tag{33}$$

The numerical calculations of $C_{2k}(\omega)$, $C_{2k}^+(\omega)$ can be performed using trapezoidal rule. The Simpson rule has not any advance because of smoothness and periodicity of integrand. The functions $\Psi(\omega \cos \varphi, \omega \sin \varphi)$, $\Psi^+(\omega \cos \varphi, \omega \sin \varphi)$ were evaluated with $\triangle \varphi = \frac{\pi}{30}$. The values of $\Phi(\omega_1, \omega_2)$, $\Phi^+(\omega_1, \omega_2)$ can be obtained directly from $F(\omega_1, \omega_2)$, $F^+(\omega_1, \omega_2)$. Alternative way of $\Phi^+(\omega_1, \omega_2)$ calculation consists of 2D FFT for $N = 2^M$ and 2D cubic spline interpolation.

5   Testing example: Circle

In the case of circle with radius $R > 0$ we obtained

$$\Phi(\omega_1, \omega_2) = \begin{cases} \frac{4J_1^2(\omega R)}{\omega^2 R^2}, & \omega > 0, \\ 1, & \omega = 0, \end{cases} \tag{34}$$

where $\omega = (\omega_1^2 + \omega_2^2)^{\frac{1}{2}}$ and J_1 is Bessel function of 1st kind. After affine transform we have

$$\Psi(\omega_1, \omega_2) = \begin{cases} \frac{J_1^2(2\omega)}{\omega^2}, & \omega > 0, \\ 1, & \omega = 0. \end{cases} \tag{35}$$

Thus, $\Psi(\omega_1, \omega_2) = 1 - \omega^2 + O(\omega^4)$. Because of radial symmetry of $\Psi(\omega_1, \omega_2)$ we obtained

$$C_0(\omega) = \begin{cases} \frac{J_1^4(2\omega)}{\omega^4}, & \omega > 0, \\ 1, & \omega = 0, \end{cases} \tag{36}$$

$$C_n(\omega) \equiv 0 \quad \text{for } n \in \mathbb{N}, \tag{37}$$

The approximation error

$$\varepsilon = \max \left| C_n^+(\omega) - C_n(\omega) \right| \tag{38}$$

was studied for various radii $R > 0$ and non-integer shifts of circle center. When $R \geq 25$ then
$\varepsilon < 0.0002$ for $\omega \in [e^{-\frac{1}{2}}, e^2]$, $n \leq 30$.

6   Experimental part

Several interesting objects were studied in the experimental part. These objects resemble each other, because of they are squares with one hole. This hole is of various shapes: rectangle, circle, diamond, triangle, hexagon, tooth and half-circle. The center of this hole is at one of three positions. Let $2a$ be size of one side of square, $s = (0,0)$ be center of this square. Then the center of hole is one of following: $s_1 = (0,0)$, $s_2 = a(\frac{1}{4}, 0)$, $s_3 = a(\frac{1}{4}, \frac{1}{3})$. Largest size of side of object inside is a.

Basic objects are in the Fig. 1. In the first line there are objects with the center s_1, in the second line with the center s_2 and in the last line with the center s_3. Then classes are formed from affine transformed basic objects. In each class there are 20 objects.
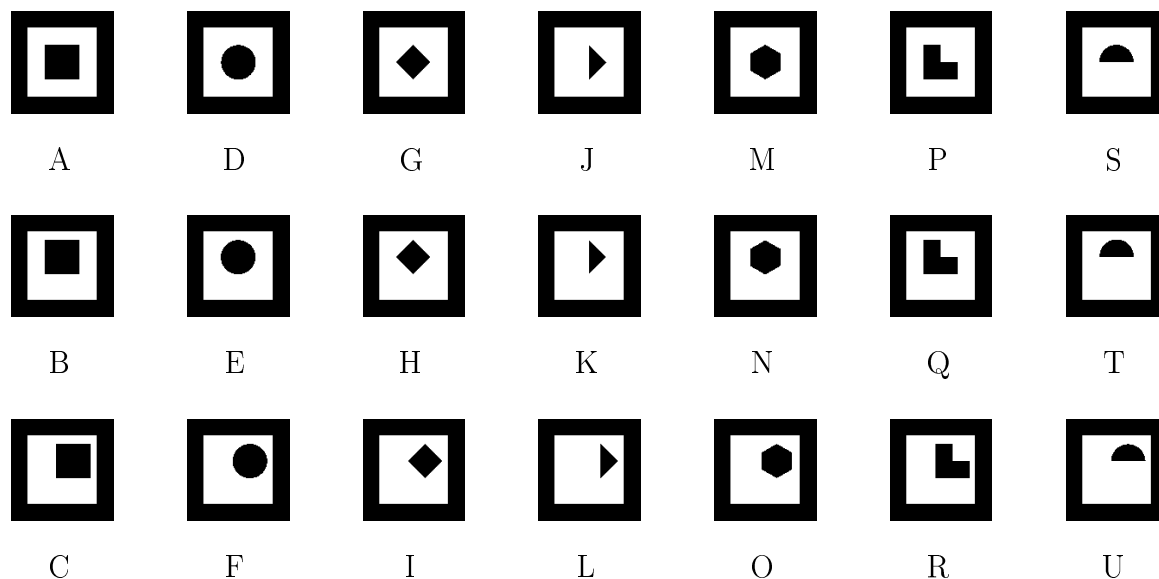

Figure 1: Class representatives

The influence of the hole shape on harmonic frequences is illustrated in the Fig. 2. There are examples of harmonics for selected objects. The legend in figures join together the colors with harmonics. The number coresponds with order of harmonic frequence. All harmonic frequences (except $C_0(\omega)$) were zoomed with factor 10 000. Then, only harmonics with the maximal value greather than 0.015 are depicted. Others don't affect the pattern recognition. The influence of the center position of hole in square on harmonics is depicted in the Fig. 3. PCA of sampled harmonics is in the Fig. 4.

7   Conclusion

The normalized 2D power spectrum $\Phi(\omega_1, \omega_2)$ was built first on the base of Fourier spectrum. The approximation of this spectrum with moments was transformed to the radial symmetric spectrum $\Psi(\omega_1, \omega_2)$. This spectrum is translation and scaling invariant. Using polar coordinates, we have $\Psi(\omega \cos \varphi, \omega \sin \varphi)$. The squared absolute values of Fourier
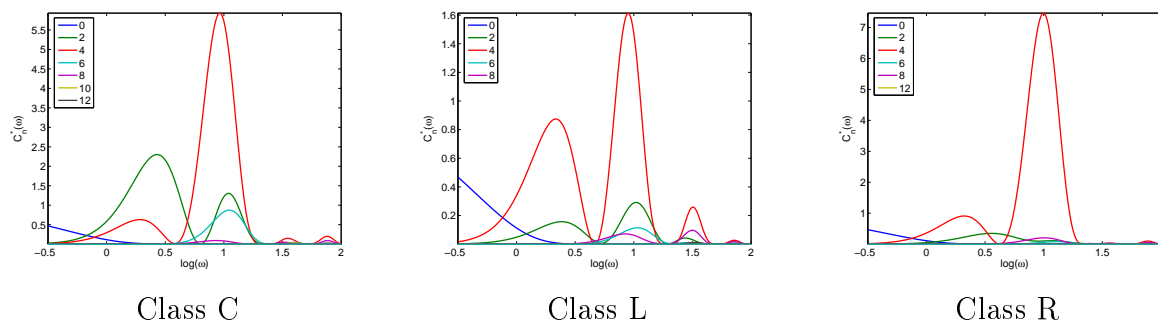
Class C   Class L   Class R

Figure 2: Harmonics for selected object classes

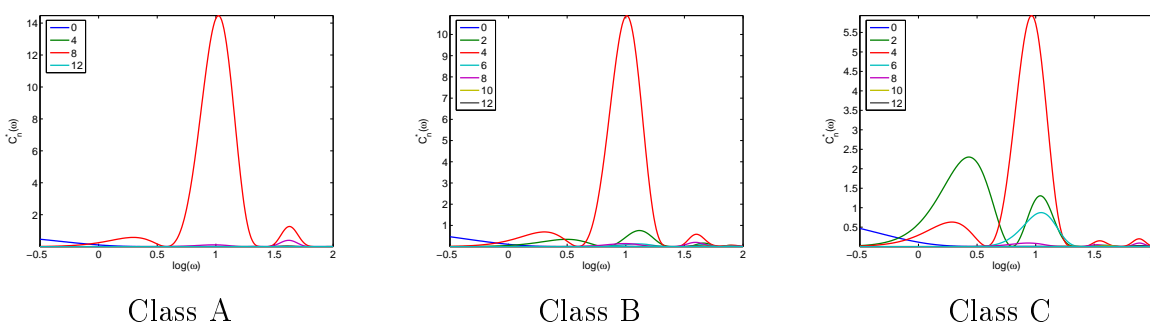Class A   Class B   Class C

Figure 3: The influence of the center position of hole in square on harmonics
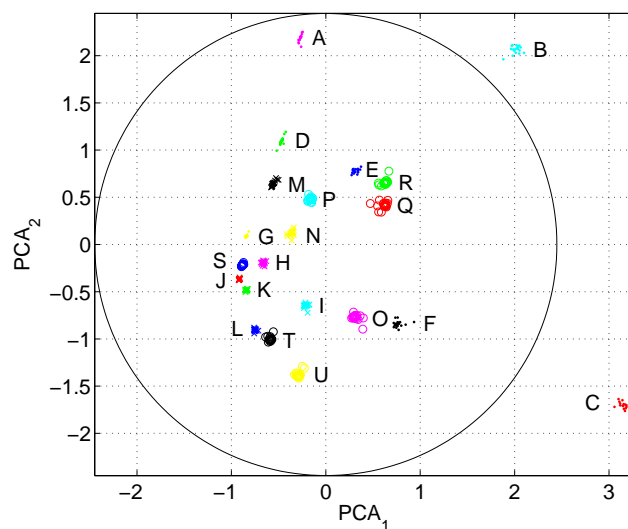

Figure 4: PCA of sampled harmonics

coefficients are $C_n(\omega)$ for $n \in \mathbb{N}_0$, $\omega \geq 0$. The functions $C_n(\omega)$ are affine invariant with respect to changes in 2D image f. The property $a \in (0,1]$ can be called inverse excentricity. In discrete case we get only approximations of these functions. The approximation error is small, which was tested in the example with the circle. Several interesting objects were studied in the experimental part. These are classes of squares with various cut out objects. Frequency dependence of Fourier coeficients $C_n(\omega)$ was studied and the adequate spectra were depicted. The effect of hole position on the harmonics was also studied. Finally, the PCA of sampled harmonics was able to separate these classes.

References

[1] R. Klette, P. Zamperoni. *Handbook of Image Processing Operators*. John Wiley & Sons, England, (2006).

[2] R.Gonzales, R. Woods. *Digital Image Processing*. Prentice-Hall, New Jersey, (2001).

Numerical Scheme for the Willmore Flow

Tomáš Oberhuber

3rd year of PGS, email: `oberhuber@kmlinux.fjfi.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Michal Beneš, Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská, ČVUT

Abstrakt. Tento článek popisuje numerické schéma pro úlohu zvanou Willmorův tok. Schéma je založeno na metodě konečných diferencí a metodě přímek. Uvádíme výsledky numerických experimentů, kdy je toto schéma použité na isotropní a anisotropní Willmorů tok pro grafy a také pro vrstevnicovou formulaci Willmorova toku.

Abstract. In this paper we present a numerical scheme for the Willmore flow. The scheme is based on the finite difference method and the method of lines. We present results of numerical experiments when the scheme is applied on isotropic and anisotropic Willmore flow of graphs and on the level-set formulation of the Willmore flow.

1 Introduction

We consider evolution of curve resp. surface $\Gamma(t)$ embedded in $\mathbb{R}^2$ resp. $\mathbb{R}^3$. We investigate the following law

$$V = 2\triangle_\Gamma H + H^3 - 4HK \; on \; \Gamma(t), \tag{1}$$

where V is the normal velocity, $\triangle_\Gamma$ is the Laplace-Beltrami operator, $H = \kappa_1 + \kappa_2$ is the mean curvature, $K = \kappa_1 \cdot \kappa_2$ is the Gauss curvature and κ_1 and κ_2 denote the principal curvatures of the surface.

As follows from [1, 2, 3] the law (1) represents the L_2-gradient flow for the functional W defined as

$$W(f) = \int_\Gamma H^2 dS. \tag{2}$$

The problem of the minimization of the functional (2) comes from the physics of elasticity [4]. In [3] the authors study evolution of elastic curves in $\mathbb{R}^n$. Application for the surface reconstruction of scratched objects is discused in [5]. We present numerical schemes for graph formulation for both isotropic and anisotropic Willmore flow of surfaces in $\mathbb{R}^3$ as well as for the level-set formulation of the Willmore flow of curves in $\mathbb{R}^2$.

2 Problem formulation

2.1 Isotropic Willmore flow of graphs

We assume that $\Gamma(t)$ is a graph of a function u of two variables:

$$\Gamma(t) = \left\{ [x, u(t, x)] \mid x \in \Omega \subset \mathbb{R}^2 \right\},$$

where $\Omega \equiv (0, L_1) \times (0, L_2)$ is on open rectangle, $\partial\Omega$ its boundary and ν its outer normal.

A graph formulation for the Willmore flow (1) is a system of two partial differential equations of the second order for u and w

$$\frac{\partial u}{\partial t} = -Q\nabla \cdot \left[\frac{2}{Q} \left(\mathbb{I} - \mathbb{P} \right) \nabla w - \frac{w^2}{Q^3} \nabla u \right] \ in \ \Omega \times (0, T) \,,$$

$$w = Q\nabla \cdot \frac{\nabla u}{Q},$$

$$u(\cdot, 0) = u_{ini},$$

with the Dirichlet boundary conditions

$$u \mid_{\partial\Omega} = g, \quad w \mid_{\partial\Omega} = 0,$$

or the Neumann boundary conditions

$$\frac{\partial u}{\partial \nu} \mid_{\partial\Omega} = 0, \quad \frac{\partial w}{\partial \nu} \mid_{\partial\Omega} = 0,$$

where

$$Q = \sqrt{1 + |\nabla u|^2}, \quad n = \frac{\nabla u}{Q}, \quad H = \nabla \cdot n, \quad \mathbb{P} = \frac{\nabla u}{Q} \otimes \frac{\nabla u}{Q}$$

and $\mathbb{I}$ is the unit matrix.

2.2   Anisotropic Willmore flow of graphs

We assume having convex and positive 1-homogeneous function

$$\gamma : \mathbb{R}^{n+1} \setminus \{0\} \to \mathbb{R}_0^+; \quad \gamma = \gamma(p_1, \cdots p_n, -1) \tag{3}$$

so called surface energy density. Denoting

$$\nabla_p \gamma = (\gamma_{p_1}, \cdots, \gamma_{p_n}) \,,$$

we define anisotropic mean curvature induced by function (3) as

$$H_\gamma = \nabla \cdot (\nabla_p \gamma (\nabla u, -1))$$

and anisotropic Willmore functional as

$$W(f) = \int_\Gamma H_\gamma^2 dS. \tag{4}$$

The graph formulation of the anisotropic Willmore flow for (4) takes the following form

$$\frac{\partial u}{\partial t} = -Q\nabla \cdot \left[2\mathbb{E}\nabla w - \frac{w^2}{Q^3} \nabla u \right],$$

$$w = QH_\gamma,$$

$$u(\cdot, 0) = u_{ini},$$

with the Dirichlet boundary conditions

$$u\mid_{\partial\Omega}= g, \quad w\mid_{\partial\Omega}= 0,$$

or the Neumann boundary conditions

$$\frac{\partial u}{\partial \nu}\mid_{\partial\Omega}= 0, \quad \frac{\partial w}{\partial \nu}\mid_{\partial\Omega}= 0.$$

where

$$\mathbb{E}_{ij} = [(\nabla_p \otimes \nabla_p)\, \gamma\,(\nabla u, -1)]_{ij} = \gamma\,(\nabla u, -1)_{p_i p_j}.$$

For the isotropic case we have

$$\gamma\,(\nabla u, -1) = \sqrt{1 + |\nabla u|^2}, \quad H_\gamma = \nabla \cdot \frac{\nabla u}{Q}, \quad \mathbb{E} = \frac{1}{Q}\left(\mathbb{I} - \frac{\nabla u}{Q} \otimes \frac{\nabla u}{Q}\right).$$

2.3 The level-set formulation for the Willmore flow

We assume having a curve in $\mathbb{R}^2$ with its interior in bounded domain Ω described as

$$\Gamma\,(t) = \{x \in \Omega \mid u(t, x) = 0\},$$

where $u(t, x)$ is smooth function defined on Ω. The level-set formulation for the Willmore flow (1) takes the following form

$$\begin{aligned} \frac{\partial u}{\partial t} &= -Q\nabla \cdot \left[2\mathbb{E}\nabla w - \frac{w^2}{Q^3}\nabla u\right], \\ w &= QH_\gamma, \\ u(\cdot, 0) &= u_{ini}, \end{aligned}$$

with the Dirichlet boundary conditions

$$u\mid_{\partial\Omega}= g \equiv c, \quad w\mid_{\partial\Omega}= 0,$$

where c is constant and

$$Q = \sqrt{\epsilon + |\nabla u|^2}, \quad \mathbb{E}_{ij} = [(\nabla_p \otimes \nabla_p)\, \gamma\,(\nabla u, -1)]_{ij} = \gamma\,(\nabla u, -1)_{p_i p_j}.$$

The initial condition u_{ini} reads

$$u_{ini}(x) = c \cdot sign\,(d_{\Gamma_0}) \left(1.0 - e^{-\left|d_{\Gamma_0}/c\right|}\right),$$

where d_{Γ_0} is the distance function for the initial curve Γ_0.

3   Numerical scheme

For the space discretisation we apply the finite difference method. Let h_1, h_2 be space steps such that $h_1 = \frac{L_1}{N_1}$ and $h_2 = \frac{L_2}{N_2}$ for some $N_1, N_2 \in \mathbb{N}^+$. We define a uniform grid as

$$
\begin{aligned}
\omega_h &= \left\{ (ih_1, jh_2) \mid i = 1 \cdots N_1 - 1, j = 1 \cdots N_2 - 1 \right\}, \\
\overline{\omega}_h &= \left\{ (ih_1, jh_2) \mid i = 0 \cdots N_1, j = 0 \cdots N_2 \right\}.
\end{aligned}
$$

For $u : \mathbb{R}^2 \to \mathbb{R}$ we define a projection on $\overline{\omega}_h$ as $u_{ij} = u(ih_1, jh_2)$. We introduce the differences

$$
u_{\mathring{x}_1, ij} = \frac{u_{i+1,j} - u_{i-1,j}}{2h_1}, \quad u_{\mathring{x}_2, ij} = \frac{u_{i,j+1} - u_{i,j-1}}{2h_2}, \quad \mathring{\nabla}_h u_{ij} = \left(u_{\mathring{x}_1, ij}, u_{\mathring{x}_2, ij} \right).
$$

For ($\epsilon = 1$ in the case of graphs)

$$
\begin{aligned}
\mathring{Q}_{ij} &= \sqrt{\epsilon + u_{\mathring{x}_1, ij}^2 + u_{\mathring{x}_2, ij}^2}, \quad \mathring{H}_{\gamma, ij}^h = \mathring{\nabla} \cdot \left(\nabla_{p_i} \gamma \left(\mathring{\nabla} u_{ij}^h, -1 \right) \right), \\
\mathring{\mathbb{E}}_{ij}^h &= (\nabla_p \otimes \nabla_p) \gamma \left(\mathring{\nabla}^h u_{ij}^h, -1 \right),
\end{aligned}
$$

the numerical scheme has the following form

$$
\begin{aligned}
\frac{du^h}{dt} &= -\overline{Q} \mathring{\nabla}_h \left(\frac{2}{\mathring{Q}} \mathring{\mathbb{E}} \mathring{\nabla}_h w^h - \frac{(w^h)^2}{\mathring{Q}^3} \mathring{\nabla}_h u^h \right), \\
w^h &= \mathring{Q} \cdot \left[\left(\frac{u^h_{\mathring{x}_1}}{\mathring{Q}} \right)_{\mathring{x}_1} + \left(\frac{u^h_{\mathring{x}_2}}{\mathring{Q}} \right)_{\mathring{x}_2} \right].
\end{aligned}
$$

with the initial condition defined as

$$
u^h(0) = u_{ini} \mid_{\overline{\omega}_h},
$$

and we consider either the Dirichlet boundary conditions

$$
u^h \mid_{\partial \omega_h} = g, \quad w^h \mid_{\partial \omega_h} = 0,
$$

or the Neumann boundary conditions

$$
u^h_{\bar{n}} \mid_{\partial \omega_h} = 0, \quad w^h_{\bar{n}} \mid_{\partial \omega} = 0.
$$

The discretisation in time is done by mean of the method of lines.

4   Numerical experiments

In this section we present results of several numerical experiments. On the Fig. 1 the initial condition is $u_0(x, y) = \sin \pi x \cdot \sin \pi y$ on the domain $\Omega = \langle 0, 1 \rangle \times \langle 0, 1 \rangle$. The initial condition for the Fig. 2 is $u_0(x, y) = 0.5 \sin \left(\pi \tanh \left(5.0 \left(x^2 + y^2 \right) - 0.25 \right) \right)$ on the domain $\Omega = \langle -1, 1 \rangle \times \langle -1, 1 \rangle$. The boundary condtions are $u \mid_{\partial \Omega} = w \mid_{\partial \Omega} = 0$. The steady
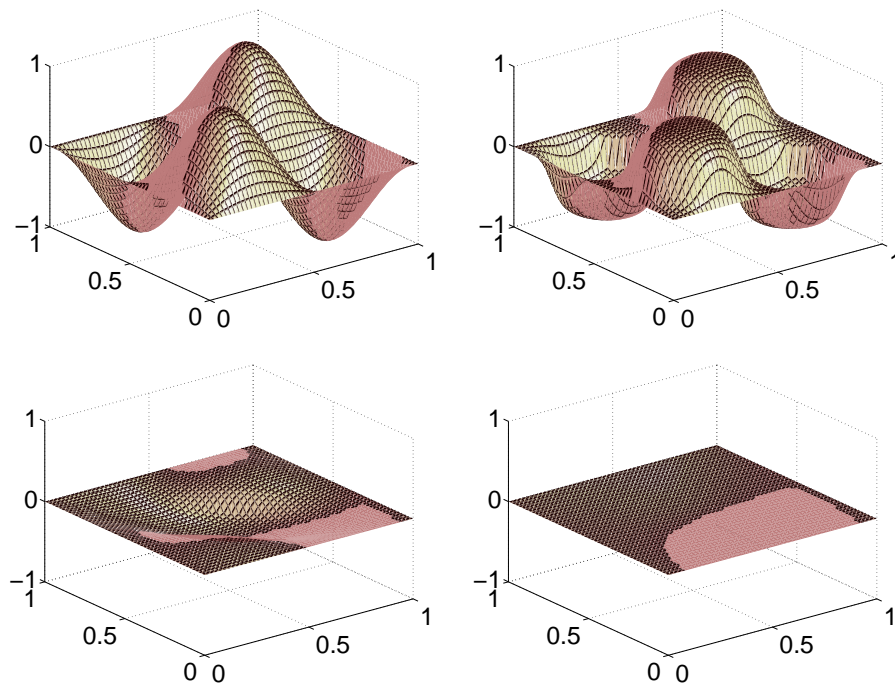
Figure 1: Decay towards the planar surface at times $t = 0$, $t = 10^{-4}$, $t = 1.7 \cdot 10^{-4}$ and $t = 0.01$.
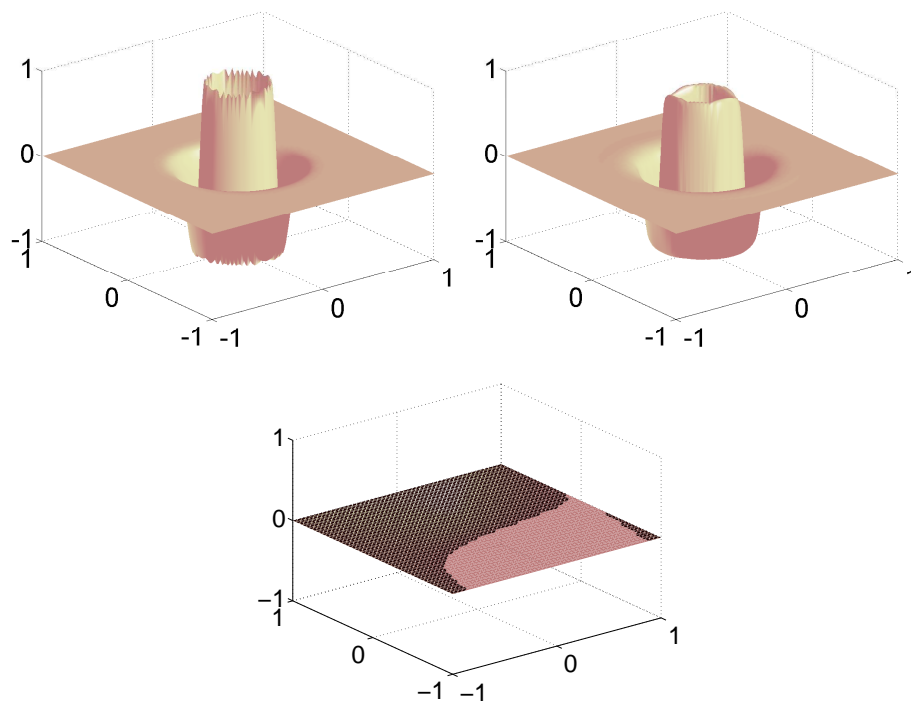

Figure 2: Decay towards the planar surface at times $t = 0$, $t = 5 \cdot 10^{-6}$ and $t = 0.1$.
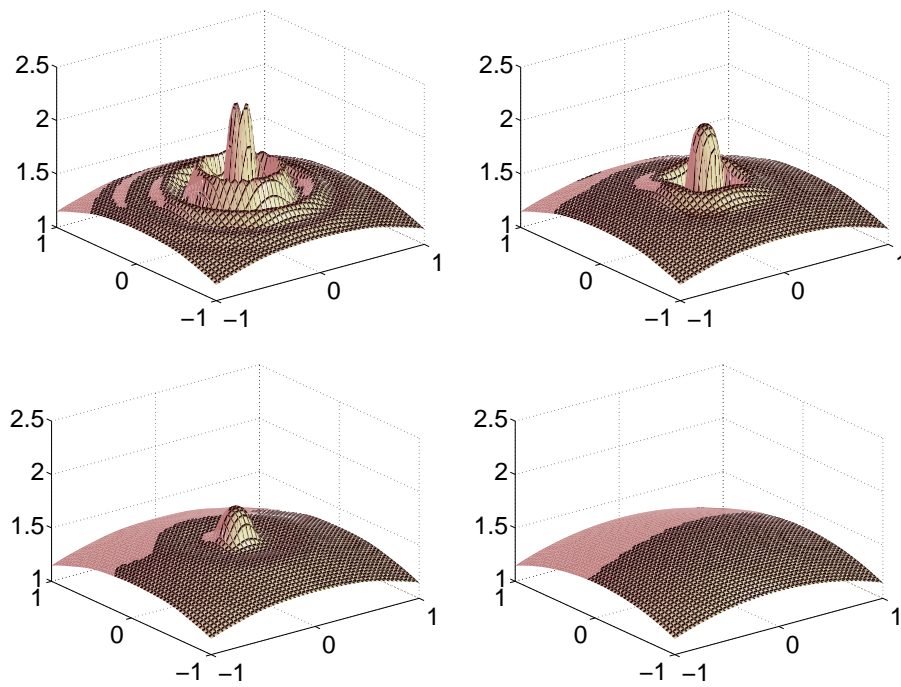
Figure 3: Spherical surface restoration at times $t = 0$, $t = 2 \cdot 10^{-5}$, $t = 10^{-4}$ and $t = 0.05$.
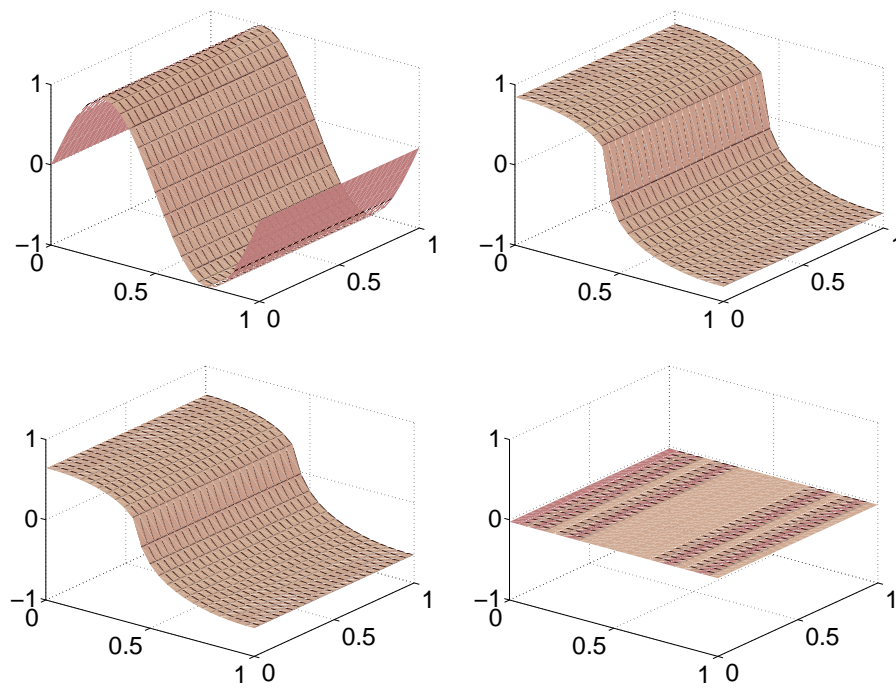

Figure 4: Test with the Neumann boundary conditions at times $t = 0$, $t = 0.005$, $t = 0.175$ and $t = 0.5$.
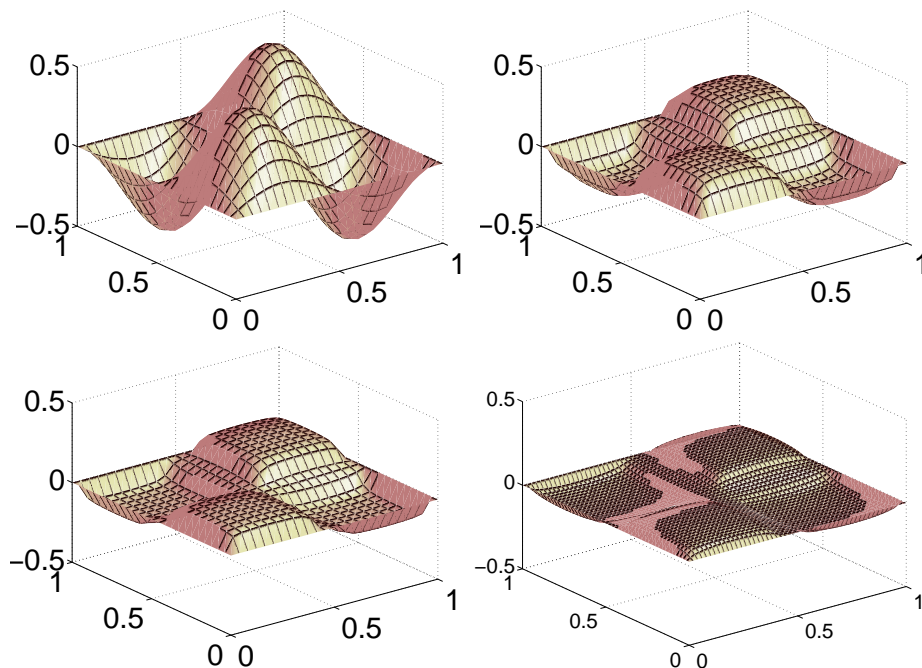
Figure 5: Convergence towards the planar surface at times $t = 0$, $t = 10^{-5}$, $t = 2 \cdot 10^{-5}$, $t = 3 \cdot 10^{-5}$ and $t = 4 \cdot 10^{-5}$.

state for both examples is the planar surface. Fig. 3 shows restoration of the spherical surface from the intial condition which is the spherical surface perturbed by function $\exp(-5\sqrt{x^2 + y^2})\sin(7.5\pi\sqrt{x^2 + y^2})$ on the domain $\Omega =< -1, 1 > \times < -1, 1 >$. The boundary conditions for u and w are computed on spherical surface with no perturbation. Fig. 4 shows evolution with the zero Neumann boundary conditions on u and w with initial condition $u_0(x, y) = \sin(\pi x)$ on the domain $\Omega =< 0, 1 > \times < 0, 1 >$. For the numerical experiments with anisotropy we consider function (3) of the form $\gamma(p, -1) = \sum_{i=1}^{3}\sqrt{P_i^2 + \epsilon\sum_{j=1}^{3}P_j^2}$ for $P_1 = p_1, P_2 = p_2, P_3 = -1$. To demonstrate the difference between isotropical and anisotropical problem the setting of the initial conditions and the boundary conditions is the same for Fig. 5 and Fig. 1 resp. Fig. 6 and Fig. 2.

Level-set Formulation for the Willmore Flow

In this section we present results of the numerical experiments with the level-set fomrulation for the Willmore flow. All of the experiments show convergence towards circles. We must note that it is not a steady state. In the case of the Willmore flow evolution of a circle radius grows to infinity.
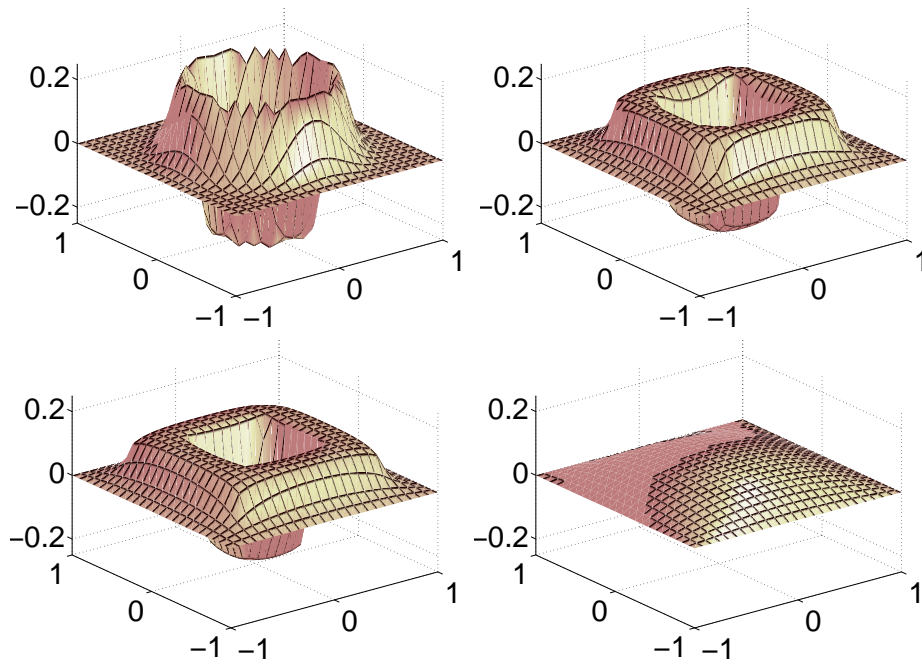
Figure 6: Convergence towards the planar surface at times $t = 0$, $t = 1.25 \cdot 10^{-4}$, $t = 2.5 \cdot 10^{-4}$ and $t = 1.25 \cdot 10^{-3}$.
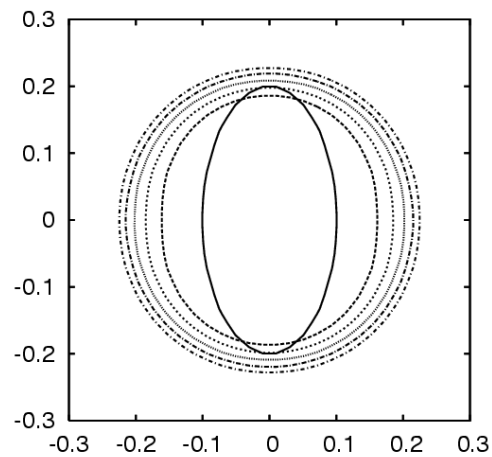

Figure 7: Ellipse growing into the circle at times $t = 0$, $t = 0.0001$, $t = 0.0002$, $t = 0.0003$, $t = 0.0004$ and $t = 0.0005$.
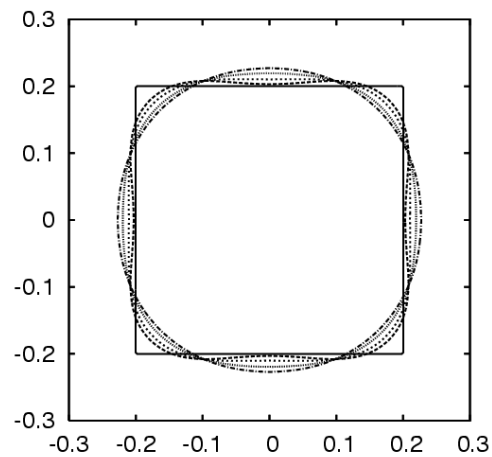
Figure 8: Square changing to the circle at times $t = 0$, $t = 2.5 \cdot e^{-6}$, $t = 5.0 \cdot e^{-6}$, $t = 1.0 \cdot e^{-5}$ and $t = 2.0 \cdot e^{-5}$.
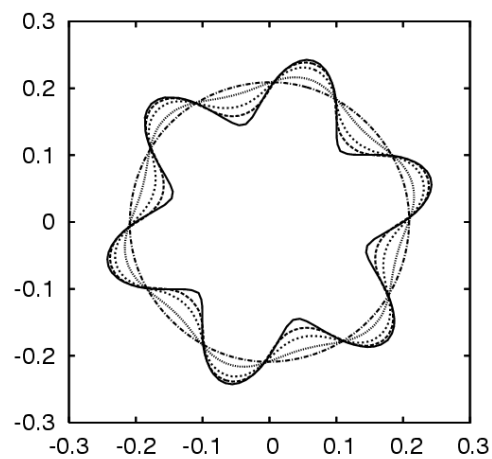

Figure 9: "Flower" changing to the circle at times $t = 0$, $t = 4.0 \cdot e^{-7}$, $t = 1.0 \cdot e^{-6}$, $t = 2.0 \cdot e^{-6}$ and $t = 5.0 \cdot e^{-6}$.
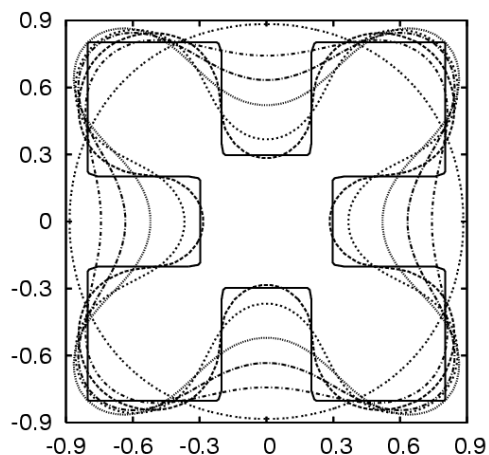
Figure 10: Squares changing to the circle at times $t = 0$, $t = 2.5 \cdot e^{-4}$, $t = 0.001$, $t = 0.0025$, $t = 0.00375$, $t = 0.005$ and $t = 0.01$.

References

[1] K. Deckelnick and G. Dziuk. *Error estimates for the Willmore flow of graphs*, In 'Interfaces and Free Boundaries 8 2006', 21–46.

[2] M. Droske and M. Rumpf. *A level set formulation for Willmore flow*, In 'Interfaces and Free Boundaries 6 2004 No. 3', 361–378.

[3] G. Dziuk, E. Kuwert and R. Schätzle. *Evolution of Elastic Curves in $\mathbb{R}^n$: Existence and Computation*, In 'SIAM J. Math. Anal. 2006 Vol. 41 No. 6', 2161–2179.

[4] P. Ciarlet, *Mathematical Elasticity, Vol III: Theory of Shells*, North-Holland, 2000.

[5] U.Clarenz, U. Diewald, G. Dziuk, M. Rumpf and R. Rusu", *A finite element method for surface restoration with smooth boundary conditions*, In 'Computer Aided Geometric Design 2004 No. 21/5', 427–445.

Problems with Multi-Resolution Visualisation of Data using Self-Organising Maps

Philip Prentis

3rd year of PGS, email: `prentisp@km1.fjfi.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical Engineering, CTU
advisor: Ladislav Andrej, Institute of Computer Science, Academy of Sciences of the Czech Republic

Abstract. This paper describes how self-organising maps can be used to visualise data at different resolutions and discusses some of the problems that may arise. The tree-structured self-organising map is proposed as a solution and different modifications of this algorithm are explored.

Abstrakt. Tento článek popisuje jak samoorganizující se mapy lze využít k visualizaci dat ve více rozlišeních a rozebírá některé problémy, které se mohou při tom vyskytnout. Jako řešení je navržena samoorganizující se mapa se stromovou strukturu a jsou zkoumány varianty tohoto algoritmu.

1 Introduction

Self-organising maps (SOM) are a type of artificial neural network that uses unsupervised learning. They were developed by Kohonen [4] in the eighties and have since been employed successfully in a number of applications; in particular they have been used for cluster analysis and visualising high-dimensional data. One example of such an application is the WEBSOM project [4, 3], which uses SOM to map, organise and browse large document files in a two-dimensional content-addressable space. A similar application for browsing and organising colour image galleries called GalSOM was developed by myself and described in [10].

In an effort to speed up large SOM and acquire mappings at different resolutions, hierarchical variants were developed. These include the Multi-Layer SOM [2], the Evolving Tree [9] and most importantly, the Tree-Structured Self-Organising Map (TS-SOM) [5, 6], which this paper describes in detail. One of the best-known applications to use TS-SOM is PicSOM [7, 8], which uses multiple TS-SOM to facilitate content-based image retrieval.

Incorporating TS-SOM into the GalSOM image browser led to an improvement of the algorithm with *multi-resolution correction* as described in [11]. More general information about using TS-SOM for image browsing and data analysis is discussed in [12]. This paper discusses the various problems that may arise when using SOM to visualise data.
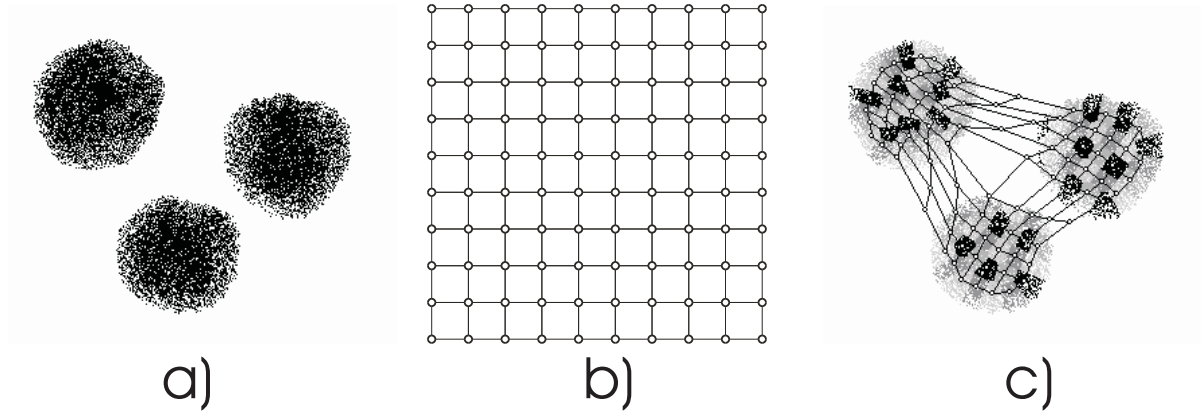
Figure 1: a) 2-dimensional input vectors in 3 clusters. b) A SOM, topologically ordered in a grid of 10×10 neurons c) The SOM adapts itself to match the input space, each neuron's codebook vector quantifying a set of inputs.

2   Self-Organizing Maps

Simply put, a SOM reduces a high-dimensional input-space (set of vectors from R^N) to a two-dimensional ordered grid of codebook vectors (neurons) $\{n_{ij}|i = 1..N_1, j = 1..N_2; \text{typically} N_1 = N_2\}$ that quantify it. In a successfully adapted map adjacent grid nodes will quantify similar data, i.e. data points that have a small Euclidean distance. The principle may be seen in figure 1, which shows a typical SOM in a 2D input space. A simple description of the basic algorithm may be found in [13].

2.1   The algorithm

The self-organisation process is achieved as follows:

1. Initialise the codebook vectors $n_{ij}(0)$ at random (usually by setting them to randomly chosen input vectors).

2. Select a random input $i(t)$ and find the best matching neuron (BMN) $n_{best}(t)$ (i.e. the neuron with the closest codebook vector). Every input sample has the same probability of being selected.

3. Move the BMN and its topological neighbours within a certain neighbourhood distance towards the selected input vector. Units located topologically further from BMN are moved less.

$$n_{ij}(t + 1) = n_{ij}(t) + \eta(t) \cdot \phi(i, j, t) \cdot [i(t) - n(t)], \tag{1}$$

where

$$\eta(t): \ N_0 \rightarrow < 0; 1 > \qquad \text{monotonously decreasing,} \tag{2}$$
$$\phi(i, j, t): \ N_0 \times N_0 \times N_0 \rightarrow < 0; 1 >,$$

ϕ decreases monotonously with the topological distance of n_{ij} from n_{best} and with t. The topological distance is the length of the shortest path from one neuron to the other in the graph (grid) that represents the network's topology.
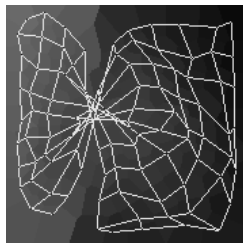
Figure 2: Due to a poorly chosen neighbourhood function, this SOM is tangled in 2D space as it attempts to adapt to a square-shaped cluster of inputs.

4. Proceed to iteration $t + 1$. Repeat 2 and 3 iteratively, reducing the proportion of the distance moved η and the neighbourhood distance ϕ each iteration, until they reach a certain predetermined threshold.

As a result the codebook vectors will be attracted to large clusters of input vectors as these will have a higher probability of being selected than sparsely populated areas of input space. η and ϕ must be selected with care if the algorithm is to achieve good results [13].

2.2 Neighbourhood function

Function ϕ in (2) is called the *neighbourhood function*. It determines how much and how distant (from the BMN) neurons will be affected at a given moment during the adaptation process. Problems can arise if the neighbourhood is decreased to soon during the adaptation process as is shown in figure 2. See [10] for in-depth heuristic analysis and optimisation.

3 Network resolution

When using a SOM to visualise a data set, it is important to choose a network of the correct size. This may be best described by the neuron/input (NI) ratio.

$$NI = N/I \tag{3}$$

where N is the number of neurons or nodes in the network and I is the number of input vectors being visualised. In this section we will show the differences between using a high or a low input ratio and discuss which are more suitable for different tasks.

3.1 Low NI ratio

For the purposes of this paper $NI < 1$ is considered low. A low NI ratio assures that most or all neurons will quantify at least some input vectors. This will hold truer the lower the ratio and in the extreme case of $NI = 1/I$ will be guaranteed. On average each neuron will quantify $1/NI$ input vectors. If used for data browsing, then we select such a ratio that $1/NI$ is the number of data elements we would typically like to browse at a

Figure 3: a) A 5×5 SOM visualising a set of bitmap images characterised by colour-histogram vectors. Each square is coloured in the mean colour of the codebook vector's histogram. b) Thumbnails of the images quantified by a selected node, in this case the centre node. Each node in the map quantifies a small easily-browsable number of similar images.

given moment. If NI is too low, the user may be confronted with too many elements at once causing him to miss the one he was looking for. If it is too high then searching may be difficult, as many of the nodes will quantify only a few elements. Nodes that quantify no elements at all are not useful because they waste the user's time and may be confusing to navigate through.

The goal of the optimisation procedure is to achieve as even a distribution as possible while preserving topology and minimising the average quantisation error (AQE). AQE is the average distance of an element in input space from its BMN. The more we demand a rigid topology, the harder it is to achieve a low AQE. Usually, good topology preserving tends towards a more even distribution [10].

If used correctly, this method allows us to view data in chunks organised by similarity into a map. It is most suitable for direct browsing. On the downside, the low network resolution does not show clusters very well and is unsuitable for analysing input space as a whole. Figure 3 shows a typical mapping.

3.2   High NI ratio

As discussed in [14], large-scale networks with many neurons may be used to bring out the emergent qualities of the SOM. We observe the overall structure of the SOM rather than the data quantified by individual neurons. One method is to calculate the average distance of each unit from its immediate topological neighbours, creating what is called a U-Matrix.

$$U \equiv [u_{ij}], \tag{4}$$

$$u_{ij} = \frac{1}{\mid X_{ij} \mid} \sum_{k,l \in X_{ij}} \mid n_{ij} - n_{kl} \mid,$$

where

$$X_{ij} \equiv \Big\{ k,l \mid k \in \{1 \ldots N_1\}, l \in \{1 \ldots N_2\}; \mid k - i \mid + \mid l - j \mid = 1 \Big\}$$

If we display a U-matrix graphically as in figure 4, we can see how clusters are being mapped by noting which areas of the matrix have low values. Graphically, these will
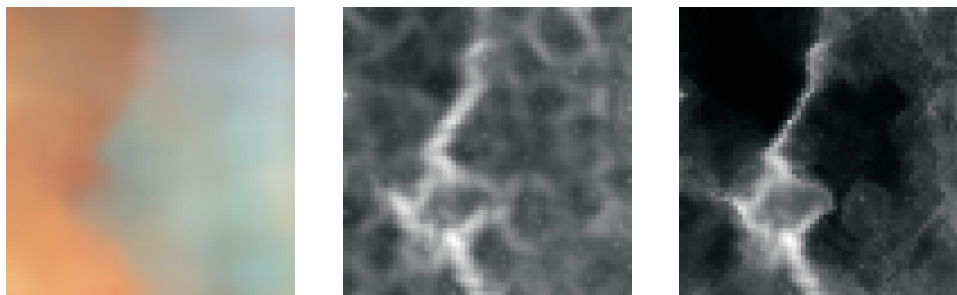
Figure 4: The colour map, U-matrix and U*-matrix of a 64 × 64 SOM mapping of 121 data elements in approximately 4 clusters. In each of these clusters further sub-clusters may be found. The U-matrix is made up of the average distances of the neurons from their immediate neighbours, black signifying very close and white very distant. Neurons within a cluster will be close to each other while at the edges they will be further apart. The U*-matrix is a U-matrix that has been processed to emphasize the clusters.

appear as 'valleys' of nodes inside clusters divided by 'ridges' of borderline nodes. For more information on U-matrices (and their improved variant, the U*-matrix) see [15].

It should be noted that with high NI ratio networks most neurons will not quantify any inputs. Thus, such networks are unsuitable for direct browsing purposes, as a user would have a difficult time finding inputs sparsely distributed across the map.

When using a U- or U*-matrix to locate clusters, we take sets of closely-packed nodes (low U-matrix values) bordered by nodes that are far from their immediate neighbours (high U-matrix values). See figure 5. These clusters, however, are unordered and do not form a regular map.

It would be possible to supply the user with information regarding adjacent clusters or alternatively, the clusters could be displayed using a quantum tree-map with zoomable navigation [1]. Figure 5-c shows a large set of images displayed in a quantum tree-map by *Photomesa* image browser.

3.3   Problems with high NI-ratio SOM

Unfortunately, larger networks take longer to adapt, as more neurons must be searched when locating the best-matching neuron (BMN). This may be an issue if the SOM is part of a user-application where adaptation speed is paramount.

3.4   Problems with multi-resolution visualisation

It is often unclear what the optimal mapping of an input space should be and SOM may adapt differently given different yet identically distributed series of input vector selections or due to minor modifications in the parameter settings. At the very least there can be no guarantee that the resulting mappings will be orientated in a specific way. Therefore, if we make multiple mappings of an input space at different resolutions using SOM of varying size, we can not expect them to match with each other. To acquire proper multi-resolution mappings it is necessary to use the tree-structured self-organising map algorithm that is described in the next section.
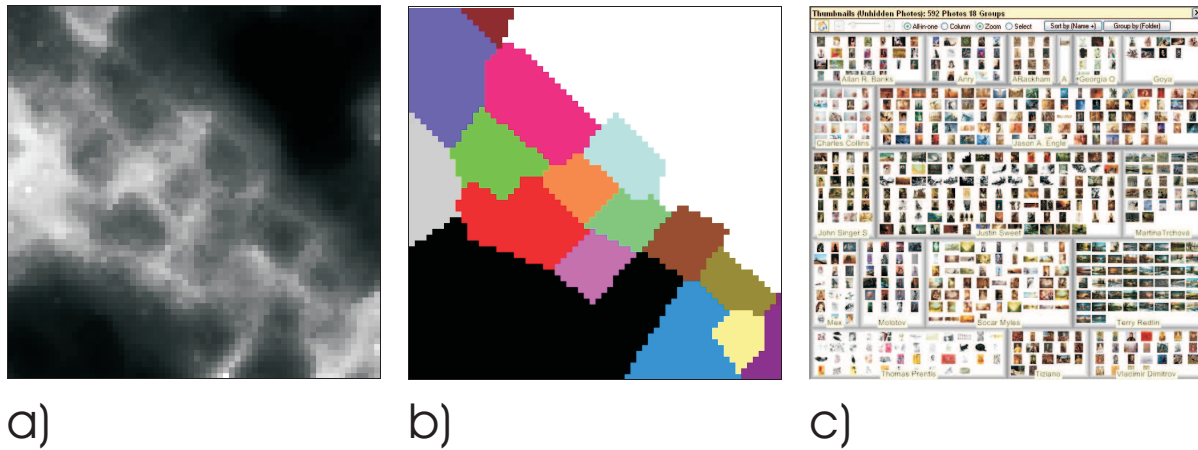
Figure 5: a) A U*-matrix b) The resulting cluster map. These clusters were selected by hand according to the borders visible on the U*-matrix. Ideally, an automatic process would be used to determine the clusters. c) A quantum tree-map such like the ones used by Photomesa image browser could be used to display the contents of the separate clusters.

4   Tree-structured self-organizing maps

One way we can simultaneously analyse input space with high and low input ratios is to use a tree-structured self-organizing map (TS-SOM), [5, 6]. This is a hierarchical structure of SOMs of exponentially increasing size. Each level of the TS-SOM adapts separately, but in the lower levels, the search for the best-matching neuron is limited to those hierarchically connected to the BMN of the previous layer. See figure 6.

4.1   The basic algorithm

The algorithm works as follows:

1. Perform one iteration of the SOM algorithm on the top layer.

2. Perform one iteration of the SOM algorithm on the next layer, but limit the search for the BMU to the neurons located under the winning neuron of the previous layer.

3. Repeat 2 until all layers have been updated.

4. Repeat 1 to 3 until the SOM thresholds have been met.

The advantages of such a structure are obvious. Instead of performing a full-search for the BMN at the lower layers, we restrict ourselves to a constant number of neurons per a given layer, thus greatly increasing the adaptation speed. The complexity of the algorithm is $O(logN)$, where N is the number of neurons on the bottom layer [6]. Also, due to the hierarchical structuring, all the SOMs will be orientated similarly in input space and the TS-SOM as a whole may be considered a multi-resolution mapping of the given data set. See figure 7.
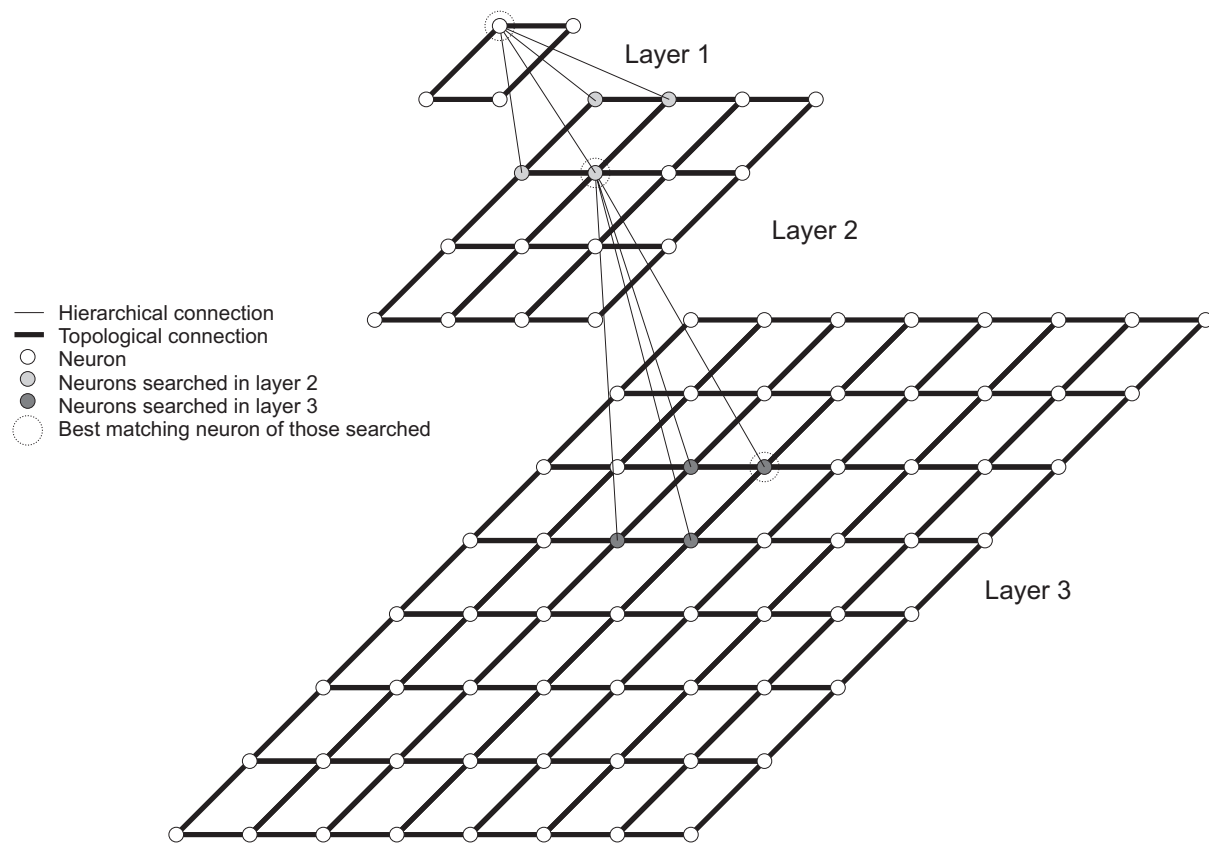
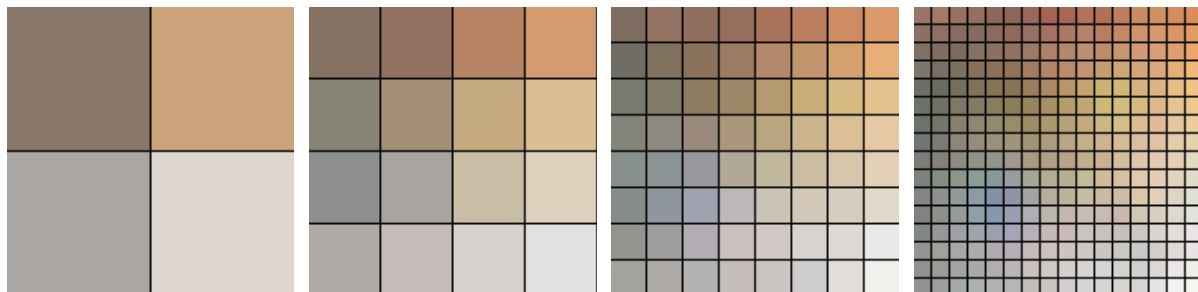Figure 6: A 3-layer TS-SOM with 4 neurons at top layer and 64 at the bottom.

Figure 7: 4 layers of a TS-SOM show colour distribution of an input space of images at different resolutions.
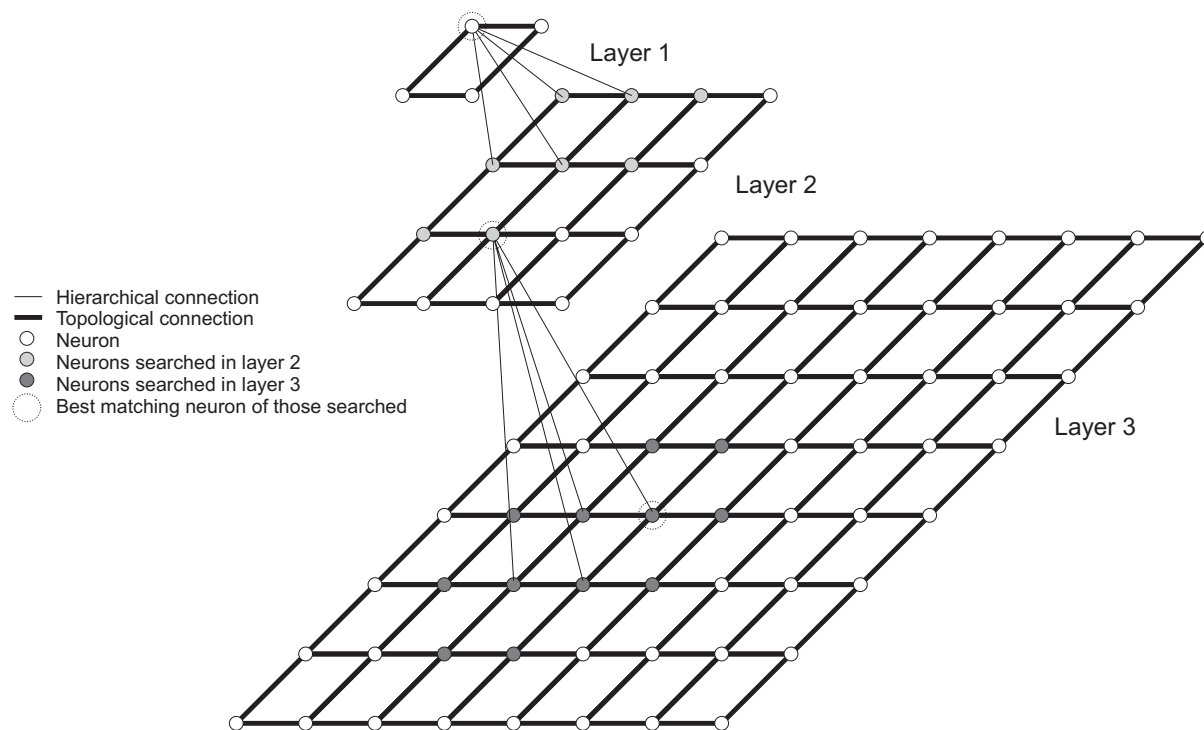
Figure 8: Wide-search for the BMN in neighbouring neurons.

Unfortunately, reducing the scope when searching for the BMN will often return suboptimal results, i.e. finding neurons that are further from the input than the closest one. As shown in my detailed analysis [11], this effect increases with each subsequent layer causing the lower high NI-ratio layers to return poor results.

4.2   Wide-search TS-SOM

The unfortunate property of the TS-SOM to propagate errors to the lower layers is caused by inputs bordering between two neurons on a higher layer, which gradually become more and more poorly quantified as the search for the BMN becomes more and more restricted. As noted in [8], better results may be achieved by allowing searching for the BMN in a wider scope, which includes neurons adjacent to those directly under a higher layer (figure 8). This is further corroborated by my experiments in [11], where I show that wide-searching is superior to the standard TS-SOM in almost all respects.

4.3   Multi-resolution correction

One unfortunate side effect of the wide-search improvement of the TS-SOM algorithm is that the separate layers become unsynchronised. This degrades the quality of the TS-SOM as a multi-resolution mapping. In [11] we describe a simple and effective method for rectifying this problem and removing the desynchronising effect. It should be noted that good synchronisation also improves the quality of the search for the BMN and brings the TS-SOM closer to the quality of the standard full-search methods.

4.4 Comparing standard and tree-structured SOM

It should be noted that a standard SOM of a given resolution will have better properties than a TS-SOM layer of the same resolution, especially if the TS-SOM has additional lower layers and uses multi-resolution correction.

4.5 Alternative models

The TS-SOM is not the only hierarchical model of the SOM. In [2] a *Multi-layer SOM* is proposed. It differs from the TS-SOM in that the higher levels adapt to the positions of the codebook vectors of the level directly beneath them. This means that there is no synchronisation between layers. Also this method does not have the benefit of the TS-SOM's acceleration when seeking the BMN.

Another model is the *Evolving Tree*, which has an indefinite structure that evolves and grows to fit the presented data. It has good data-fitting qualities and retains the speed of the TS-SOM, but due to the unpredictable nature of its structure is unsuitable for direct-browsing applications as described in this paper. See [9].

5 Corollary and conclusions

In this paper we showed how self-organising maps (SOM) may be used to visualise data sets at different resolutions. Low resolution mappings are suitable for direct browsing methods, whereas high resolution ones may be used for cluster analysis by using U-matrices. High resolution matrices are problematic because their large size makes them slow to use. If more than one mapping is made at various resolutions for multipurpose applications, then it becomes necessary to take into account that there is no guarantee that the mappings will be identically orientated. These problems may be solved by using a tree-structured self-organising map (TS-SOM), which allows multi-resolution visualisation of the data including fast high-resolution mappings. For better synchronisation of the mappings, multi-resolution correction described in the author's work [11] is recommended.

References

[1] B. Bederson *PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps*, in proc. of *UIST 2001, ACM Symposium on User Interface Software and Technology, CHI Letters*, 3(2), pp. 71-80. 2001.

[2] J. Koh, M. Suk and S. M. Bhandarkar, *A Multilayer Self-Organizing Feature Map for Range Image Segmentation, Neural Networks*, 8:67-86, 1995.

[3] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero, and A. Saarela, *Self Organization of a Massive Document Collection, IEEE Transactions on Neural Networks*, Special Issue on Neural Networks for Data Mining and Knowledge Discovery, volume 11, number 3, pages 574-585. May 2000

[4] T. Kohonen, *Self-Organizing Maps*, Third Edition, Springer-Verlag Berlin 2001.

[5] P. Koikkalainen, E. Oja, *Self-organizing hierarchical feature maps*, Proceedings of *International Joint Conference on Neural Networks*, San Diego, CA, 1990; II:279-284.

[6] P. Koikkalainen, *Progress with the tree-structured self-organizing map*, *11th European Conference on Artificial Intelligence*, August 1994.

[7] J. Laaksonen, M. Koskela and E. Oja, *Application of Self-Organizing Maps in Content Based Image Retrieval*, in Proceedings of *ICANN'99*, Edinburgh, 1999.

[8] J. Laaksonen, M. Koskela, S. Laakso and E. Oja, *Self-Organizing Maps as a Relevance Feedback Technique in Content Based Image Retrieval*, *Pattern analysis & Applications*, 4(2-3): 140-152, June 2001.

[9] J. Pakkanen, J. Iivarinen and E. Oja, *The Evolving Tree - A Novel Self-Organizing Network for Data Analysis*, *Neural Processing Letters 20*, 2004.

[10] P. Prentis, *Associative Sorting of Data and Kohonen Neural Networks* (original title - *Asociativní třídění dat a kohonenovské neuronové sítě*), Master's thesis. Faculty of Nuclear Science and Physical Engineering, Czech Technical University in Prague. June 2004.

[11] P. Prentis *Multi-Resolution Visualisation of Data with Self-Organizing Maps*, submitted to *Neural Network World* for publication, 2006.

[12] P. Prentis *Multi-Resolution Visualisation of Image Sets with Self-Organizing Maps*, in proceedings of *Doktorandský den Ústavu informatiky, Akademie věd České republiky*, 2006.

[13] R. Rojas, *Neural Networks - A Systematic Introduction*, Springer-Verlag Berlin 1996.

[14] A. Ultsch, *Data Mining and Knowledge Discovery with Emergent Self-Organizing Feature Maps for Multivariate Time Series*, In E. Oja and S. Kaski, editors, *Kohonen Maps*, pages 33–45, Amsterdam, 1999. Elsevier.

[15] A. Ultsch, *Clustering with SOM: U*C*, in Proceedings of the *5th Workshop on Self-Organzing Maps*, Paris 2005.

[16] The WEBSOM research group, *WEBSOM - Self-Organizing Maps for Internet Exploration*, `http://websom.hut.fi/websom/`.

Numerical Solution of a Flow over a Hill

Kateřina Seinerová

3rd year of PGS, email: `seinerova@centrum.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Karel Kozel, Department of Technical Mathematics, Faculty of Mechanical Engineering, Czech Technical University in Prague

Abstract. This paper deals with a numerical solution of a viscous flow over a two-dimensional hill. The mathematical model is based on the system of Navier-Stokes equations. Space discretization is done by FDM, time discretization by multistage Runge-Kutta method. To compute pressure in time, artificial compressibility method and time-marching method are used. Results of flows over a hill are presented.

Abstrakt. Tento článek pojednává o numerickém řešení vazkého proudění přes dvoudimenzionální kopec. Matematický model je založen na systému Navierových-Stokesových rovnic. Prostorová diskretizace je provedena metodou sítí, časová diskretizace vícestupňovou metodou a metodou časového ustalování. Prezentovány jsou výsledky proudění přes kopec.

1 Introduction

Flows over topography have attracted great interest from not only fluid mechanics but also engineering in a variety of fields. We consider two-dimensional internal waves excited topographically in stably flows in ABL and inthe channel of finite depth. Under these conditions, the fluid layer is bounded above by a horizontal rigid lid and below by a two-dimensional surface-mounted obstacle.

2 Solved Problems

2.1 Flows over a hill in Atmospherical Boundary Layer

The model is based on Navier-Stokes equations. Governing equations modified according to the method of artificial compressibility can be re-casted in the conservative, non-dimensional and vector form.

$$\widetilde{R}W_t + F_x + G_y = \tfrac{\widetilde{R}}{Re}(W_{xx} + W_{yy}),$$

where $W = (p, u, v)^T$ is the vector of unknowns: the pressure p and the velocity vector $(u, v)^T$. $Re = \frac{U_\infty L}{\eta/\rho} = \frac{U_\infty L}{\nu}$ is Reynolds number, $\widetilde{R} = diag(0, 1, 1)$. Terms F, G denote the inviscid fluxes

$$F = (u, u^2 + p, uv)^T,$$
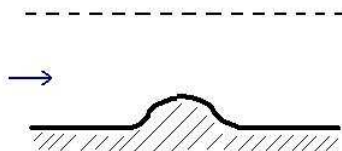
$$G = (v, uv, v^2 + p)^T.$$

Boundary conditions - normalized

entrance: $u = 1, v = 0$, extrapolated pressure $\frac{\partial p}{\partial x} = 0$

exit: extrapolation $\frac{\partial p}{\partial x} = 0, \frac{\partial u}{\partial x} = 0, \frac{\partial v}{\partial x} = 0$

upper boundary: $\frac{\partial u}{\partial y} = 0, v = 0$, extrapolated pressure: $\frac{\partial p}{\partial y} = 0$

bottom: $u = 0, v = 0$, pressure interpolated: $\frac{\partial p}{\partial n} = 0$

2.2 Flows over a hill in a channel of finite depth

This problem is described by system of Navier-Stokes equations similar to the previous problem. Nevertheless in this problem is together with the velocity (u_1, u_2) and the pressure p computed the density ρ, ρ' is the density perturbation.

$\frac{\partial u_j}{\partial x_j} = 0$

$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_j} + \frac{1}{Re} \frac{\partial^2 u_i}{\partial x_j^2} - \frac{\rho \delta_{i2}}{Fr^2}$, $i = 1, 2$

$\frac{\partial \rho'}{\partial t} + u_j \frac{\partial \rho'}{\partial x_j} = u_3$

Boundary conditions - normalized

entrance: $u_1 = 1, u_2 = 0$, extrapolated pressure $\frac{\partial p}{\partial x_1} = 0$, $\rho' = 0$

exit: extrapolation $\frac{\partial p}{\partial x_1} = 0, \frac{\partial u_1}{\partial x_1} = 0, \frac{\partial u_2}{\partial x_1} = 0, \frac{\partial \rho'}{\partial x_1} = 0$

upper boundary: $\frac{\partial u_1}{\partial x_2} = 0, u_2 = 0$, extrapolated pressure: $\frac{\partial p}{\partial x_2} = 0$, $\rho' = 0$

bottom:
bottom except of the hill - free-slip conditions $\frac{\partial p}{\partial n} = 0, u_2 = 0, \frac{\partial p}{\partial x_2} = 0, \frac{\partial u_1}{\partial x_2} = 0, \rho' = 0$
on the hill surface - no-slip condition $u_1 = 0, u_2 = 0$, interpolation: $\frac{\partial p}{\partial n} = 0, \frac{\partial \rho'}{\partial x_2} = 0$

3   Used Methods

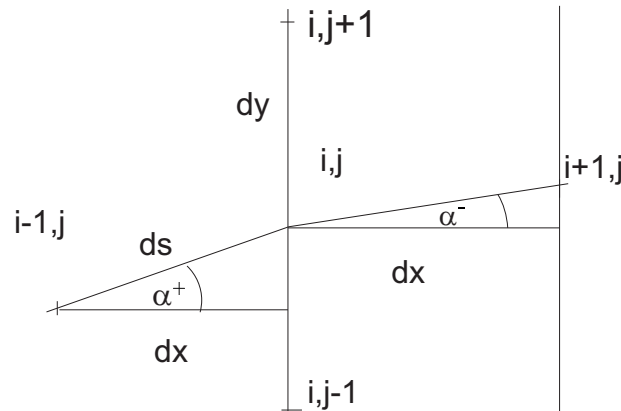## 3.1   Finite Difference Method

To discretize the governing system of equations in space, a non-orthogonal structured boundary following grid is constructed. Grid layers form an orthogonal system; the non-orthodonality is introduced to the grid by the curved terrain profile. Finite difference discretization is developed using Taylor expansion for derivatives.

Approximation of the 1st derivative in orthogonal grid: $\frac{\partial U}{\partial x} \sim \frac{1}{2}\left[\frac{U_{i+1}-U_i}{x_{i+1}-x_i} + \frac{U_i-U_{i-1}}{x_i-x_{i-1}}\right]$

After transformation of coordinates: $D_x U_{ij} = \frac{U_{i+1,j}-U_{i-1,j}}{2.\triangle s.cos\alpha} - \frac{sin\alpha[U_{i,j+1}-U_{i,j-1}]}{2.\triangle y.cos\alpha}$

Approximation of the 2nd derivative in orthogonal grid: $\frac{\partial^2 U}{\partial x^2} \sim \frac{2}{x_{i+1}-x_{i-1}}\left[\frac{U_{i+1}-U_i}{x_{i+1}-x_i} - ...\right.$
$... \left. - \frac{U_i-U_{i-1}}{x_i-x_{i-1}}\right]$

After transformation of coordinates: $D_{xx}U_{ij} = \frac{U_{i+1,j}-2U_{ij}+U_{i-1,j}}{(\triangle s.cos\alpha)^2} - ...$
$... - \frac{2sin\alpha[U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}+U_{i-1,j-1}]}{4.\triangle s.\triangle y.cos^2\alpha} + \frac{sin^2\alpha[U_{i,j+1}-2U_{ij}+U_{i,j-1}]}{(\triangle y.cos\alpha)^2} + ...$
$... + \frac{U_{i+1,j}-U_{i-1,j}}{4.\triangle s^2.cos\alpha}\left[\frac{1}{cos\alpha_i^+} - \frac{1}{cos\alpha_i^-}\right] - \frac{sin\alpha[U_{i+1,j}-U_{i-1,j}]}{4.\triangle s.\triangle y.cos\alpha}\left[\frac{1}{cos\alpha_j^+} - \frac{1}{cos\alpha_j^-}\right] - ...$
$... - \frac{U_{i,j+1}-U_{i,j-1}}{4.\triangle y.\triangle s.cos\alpha}\left[\frac{sin\alpha_i^+}{cos\alpha_i^+} - \frac{sin\alpha_i^-}{cos\alpha_i^-}\right] + \frac{sin\alpha[U_{i,j+1}-U_{i,j-1}]}{4.cos\alpha.(\triangle y)^2}\left[\frac{sin\alpha_j^+}{cos\alpha_j^+} - \frac{sin\alpha_j^-}{cos\alpha_j^-}\right]$

3.2   Multistage Runge-Kutta Method

For time discretization, 3-stage explicit Runge-Kutta method is used, especially developed for partial differential equations: $W = (p, u, v)^T$ or $W = (p, u, v, \rho)^T$

$$W_{ij}^{(0)} = W_{ij}^n$$
$$W_{ij}^{(m)} = W_{ij}^{(0)} - \alpha_m \triangle t RW_{ij}^m, m \in 1,2,3$$
$$W_{ij}^{n+1} = W_{ij}^{(3)}$$
$$RW = \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} - \frac{\tilde{R}}{Re}(\triangle W)$$

3.3   Artificial Compressibility and Time-marching Method

The solution procedure used in our method to resolve pressure is based on the so called artificial compressibility method. This requires to add the time derivative of the pressure to the continuity equation. The resultant system is then solved by time-marching method. The governing system of equations is solved in the computational domain under stationary boundary conditions for $t \to \infty$ (t is artificial time) to obtained the expected steady-state solution for the pressure and the velocity components.

4   Results

In this section, results of numerical computation, which were done on grid 250 x 100 with $Re = 2000$, are presented. Examples differ in boundary and initial conditions and in used systems of equations.

Isolines of velocity and pressure are shown in figures (1) and (2). These examples differ in boundary conditions: (1) no-slip conditions on the hill surface, free-slip conditions on the rest of surface, (2) no-slip conditions on the whole surface.

As was said, in case of computation of flows in a channel, the density is computed. Figure (3) and (4) demonstrate examples of the density perturbation $\rho' = \rho - \rho_B$ and the density ρ. Results in both of these figures differ in value of $\frac{\partial \rho_B}{\partial y}$.

Figure (5) deals with residua of presented examples to show the convergence of used methods. The first one is the residuum of the computation of flows in a channel; the second one belongs to the computation of flows in ABL; the last one shows the residuum belonging to flows in a channel again, but now $\frac{\partial \rho_B}{\partial y}$ is 2.5 times greater than in the first case.

References

[1] Uchida, T. and Ohya, Y.: Numerical study of stably stratified flows over a two-dimensional hill in a chanel of finite depth. *Fluid Dynamics Research*, vol. 29, (2001), 227–250.

[2] Bodnár, T.: *Numerical Simulation of Flows and Pollution Dispersion in Atmospheric Boundary Layer*. Disertation Thesis, Prague, (2003).

[3] Kozel, K. - Dvořák, R.: *Mathematical methods in aerodynamics*. Vydavateltví ČVUT, Praha, 1992.
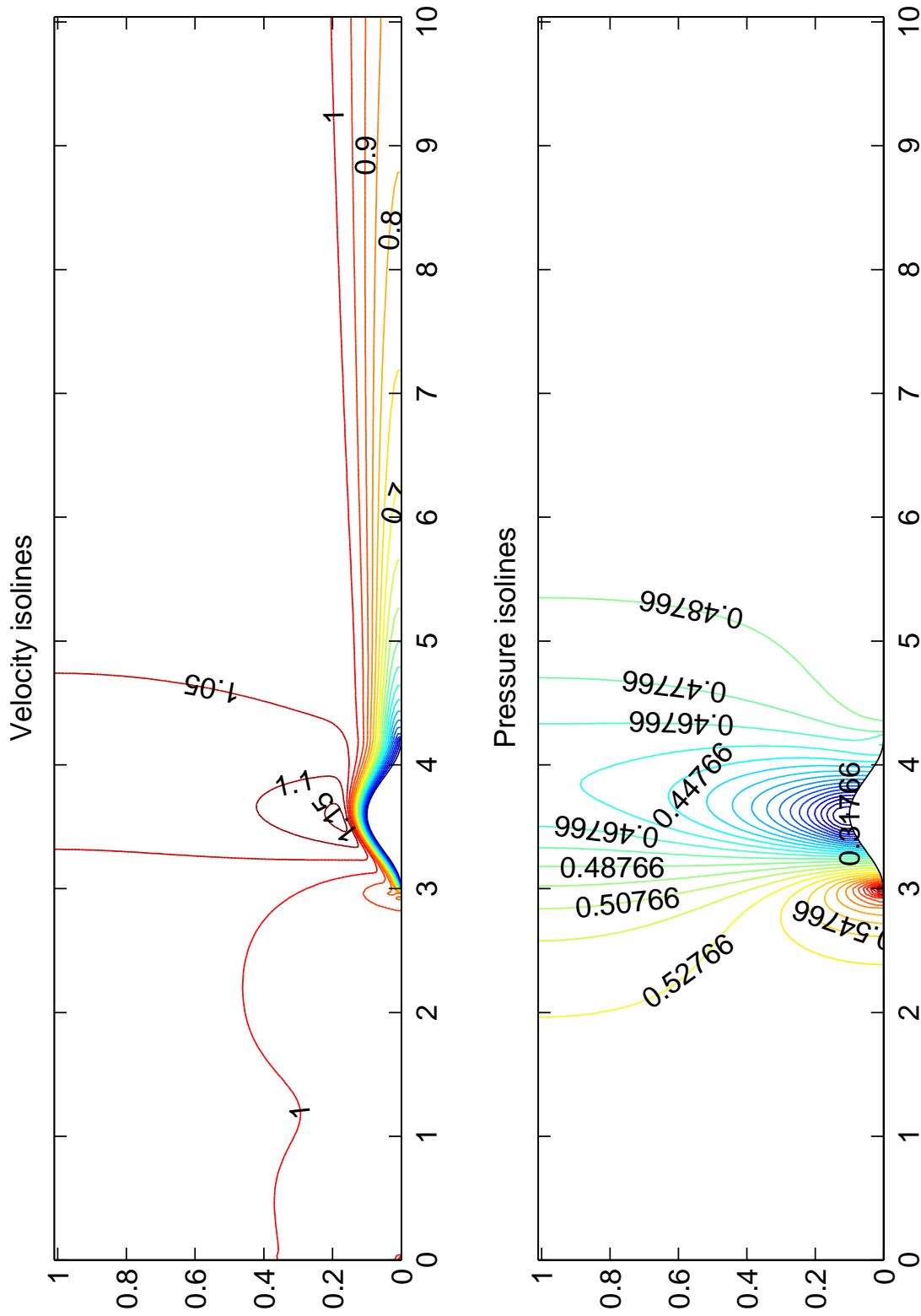
Figure 1: Isolines of velocity and pressure for flow in a channel of finite depth, $Re = 2000$, 10000 iterations.
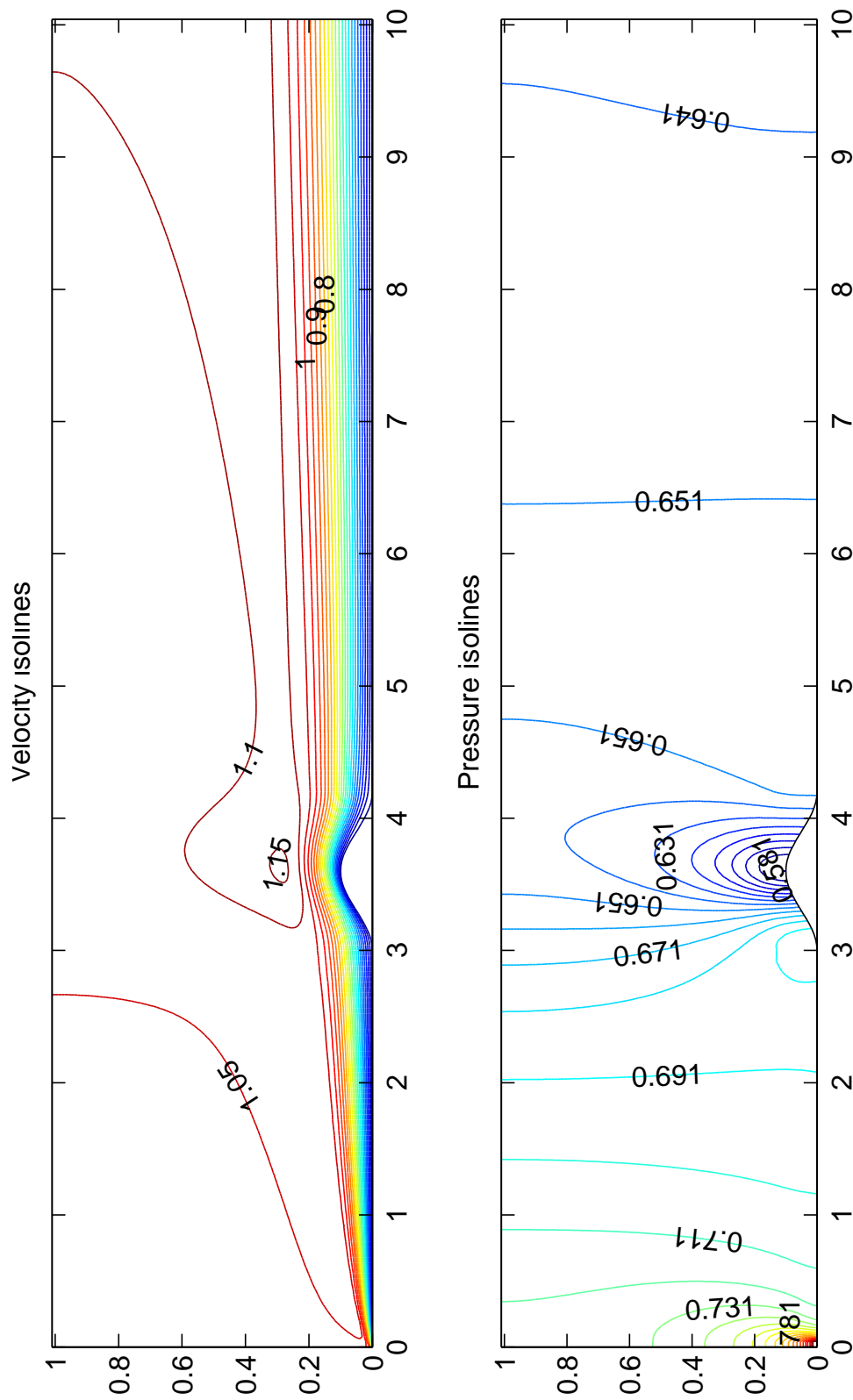
Figure 2: Isolines of velocity and pressure for flow in ABL, $Re = 2000$, 10000 iterations.
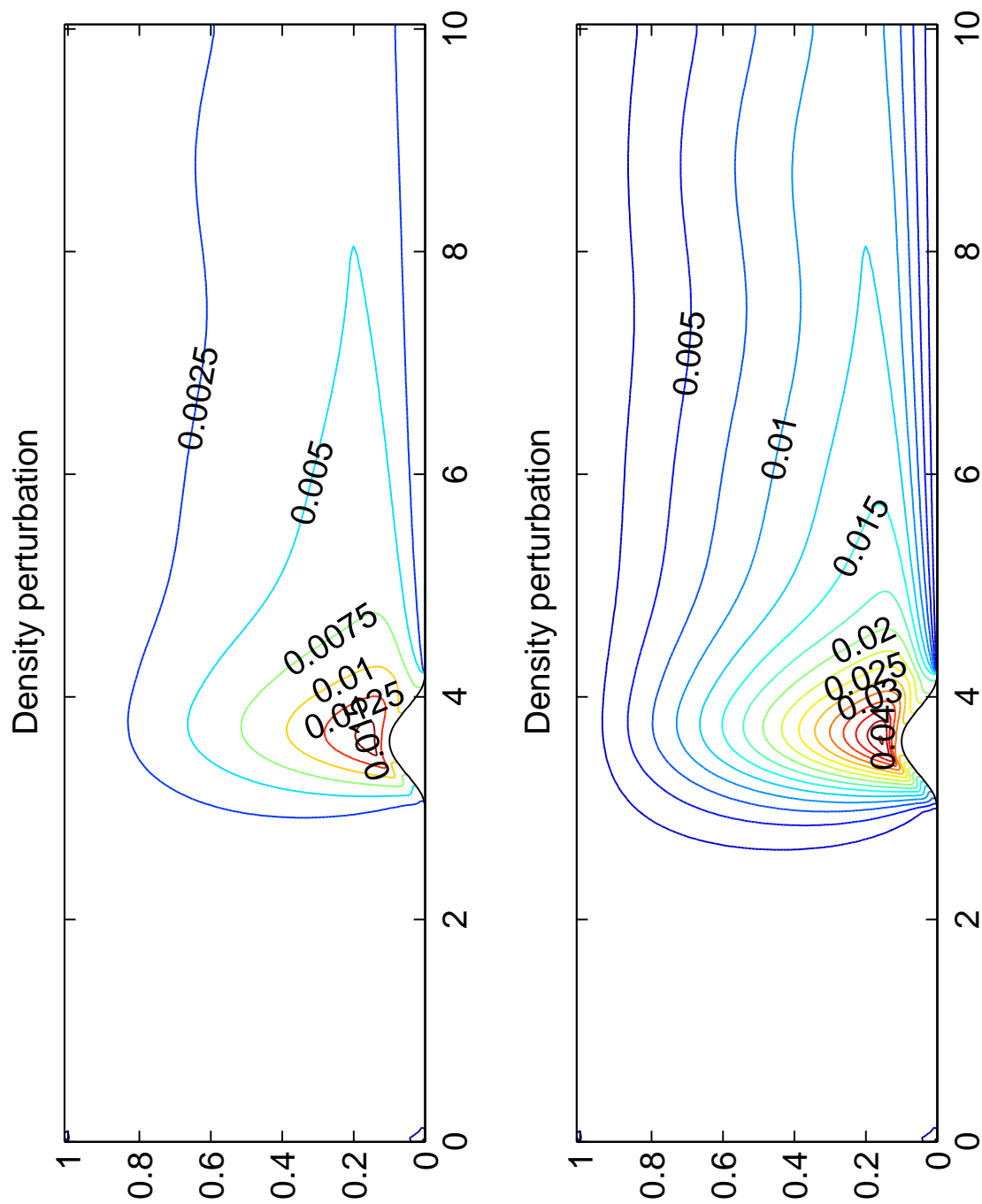
Figure 3: Isolines of density perturbation, flow in a channel of finite depth, $Re = 2000$, 10000 iterations. In the second case, $\frac{\partial \rho_B}{\partial y}$ is 2.5 times greater than in the first case.
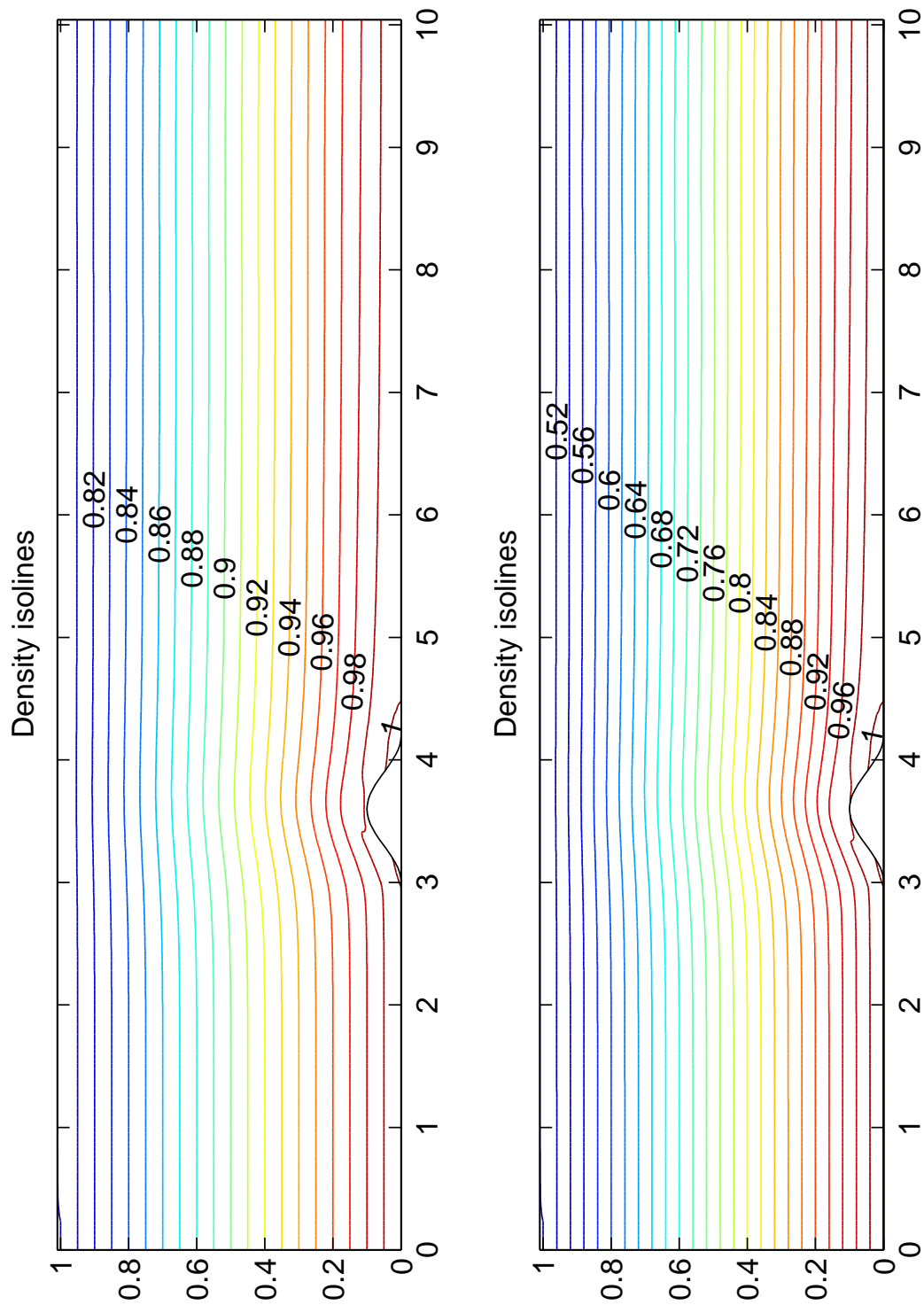
Figure 4: Isolines of density, flow in a channel of finite depth, $Re = 2000$, 10000 iterations. In the second case, $\frac{\partial \rho_B}{\partial y}$ is 2.5 times greater than in the first case
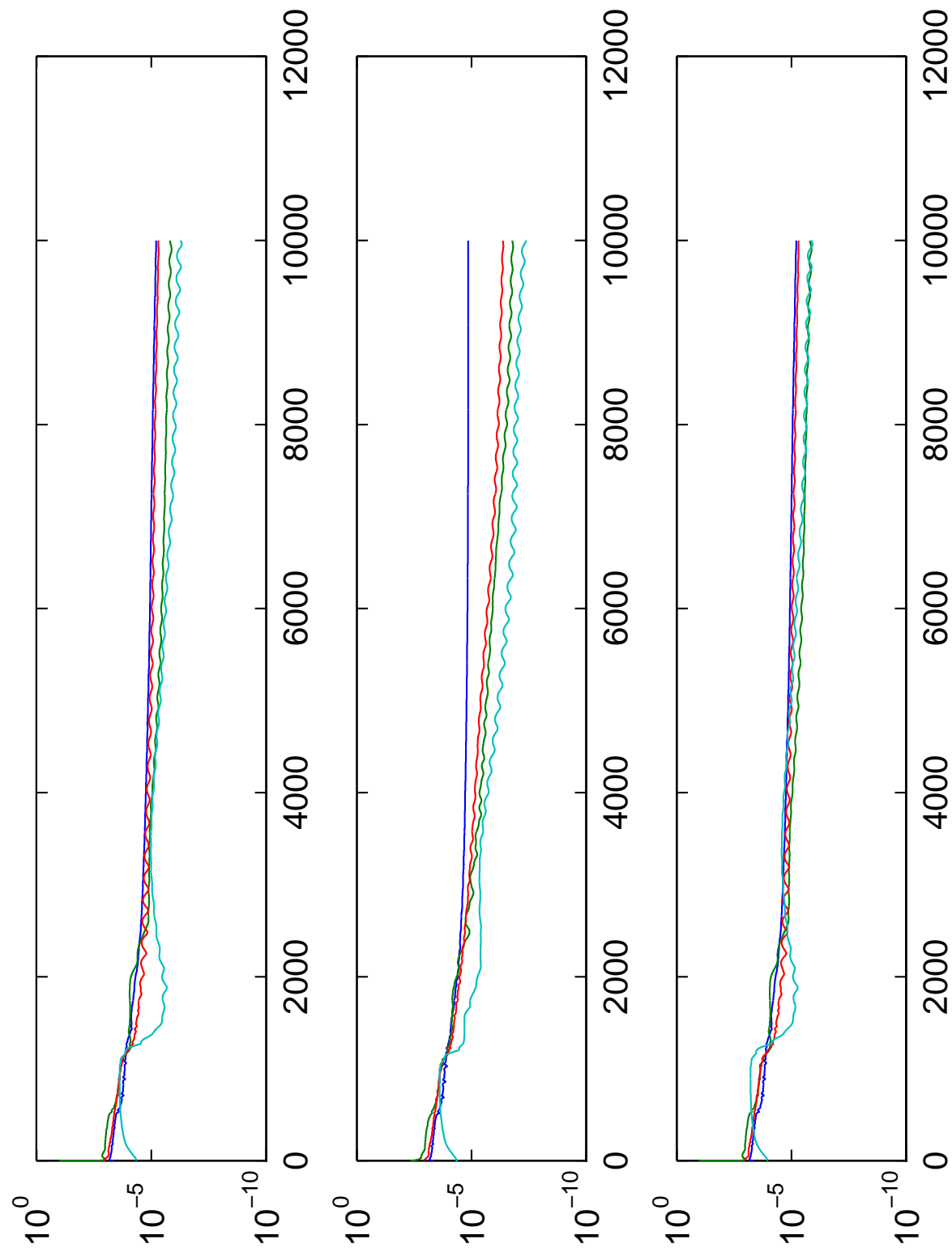
Figure 5: Residua. 1. Flows in a channel, $\frac{\partial \rho_B}{\partial y} = C$. 2. Flows in ABL, $\frac{\partial \rho_B}{\partial y} = C$. 3. Flows in a channel, $\frac{\partial \rho_B}{\partial y} = 2.5C$

Finite-Volume Model of Pulverized Coal Combustion[*]

Robert Straka[†]

2nd year of PGS, email: `straka@kmlinux.fjfi.cvut.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Michal Beneš, Department of Mathematics, Faculty of Nuclear Sciences and Physical Engineering, CTU

Abstract. We describe behavior of the burning air-coal mixture in power plant furnace, using the Navier-Stokes equations for gas and particle phases, accompanied by a turbulence model. The undergoing chemical reactions are described by the Arrhenian kinetics (reaction rate proportional to $\exp\left(-\frac{E}{RT}\right)$, where T is temperature). We also consider the heat transfer via conduction and radiation. The system of PDEs is discretized using the finite volume method (FVM) and an advection upstream splitting method as the Riemann solver. The resulting ODEs are solved using the 4th-order Runge-Kutta method. Sample simulation results for typical power production levels are presented.

Abstrakt.

Popisujeme chování hořící směsi práškového uhlí a vzduchu v elektrárenském kotli pomocí Navier-Stokesových rovnic pro plynnou a pevnou fázi spolu s modelem turbulence. Modelované chemické reakce se řídí tzv. Arrheniovskou kinetikou (rychlost reakce je uměrná $\exp\left(-\frac{E}{RT}\right)$, kde T je teplota). Dále uvažujeme přenos tepla vedením a radiací. Systém parciálních diferenciálních rovnic je diskretizovám metodami konečných objemů a "advection upstream splitting". Výsledný systém obyčejných diferenciálních rovnic je řešen metodou Runge-Kutty 4. řádu. Jsou prezentovány výsledky simulací pro typické úrovně výkonu elektrárny.

1 Introduction

The main motivation of the combustion model research is its future inclusion in the current model of the steam generator [4], to use this model for development of control systems for the industrial installation. Another purpose for the combustion model is the optimization of the production of the nitrogen oxides, which strongly depends on the temperature distribution, and thus can be controlled by intelligent distribution of fuel and oxygen into the burners. Because the experiments on a real device are prohibitively cumbersome and expensive, in extreme cases even hazardous, the only way to test the behavior of the furnace is mathematical modeling.

An industrial pulverized coal furnace is basically a vertical channel with square cross-section. The dimensions are determined by the power generation requirements from the order of meters to tens of meters. In the case we model, the furnace has 30 meters in

[†]Joined work of Jindřich Makovička, Michal Beneš and Vladimír Havlena

height and 7 meters in width, 49 m^2 cross-section. Power production of such a furnace is about 90 MW, and the furnace coupled with a steam generator is capable of producing about 100 tons of pressurized superheated steam per hour.

In the bottom of the channel walls, there are several burners — jets where the mixture of the air and coal powder is injected. The mixture then flows up and burns, while it transfers some of the combustion heat to the walls containing the water pipes.

At the top, the heated flue gas continues to flow to the superheater channel where further heat exchange occurs, and this has already been covered by [4]. Our main concern is now modeling of the processes in the area, where the coal gets burnt and nitric oxides are produced.

2   Mathematical model

The mathematical model of combustion is based on the Navier-Stokes equations for a mixture of multiple components where the coal particle are treated as one of the phases. Unlike e.g. in [1], where the gas particles are treated separately and use separate equations of momentum, we chose to use this approach, as it simplifies the model especially when dealing with turbulence, and also removes several empirical relations and constants.

Currently, the following components of the mixture are considered:

- chemical compounds engaged in major thermal and fuel NOx reactions: nitrogen (N_2), oxygen (O_2), nitric oxide (NO), hydrogen cyanide (HCN), ammonia (NH_3), and water (H_2O)

- char and volatile part of the coal particles

The gas phase is described by the following equations. As stated above, the mass balance is described by equations of mass balance of each subcomponent (the Einstein summation is used)

$$\frac{\partial}{\partial t}(\rho Y_i) + \frac{\partial}{\partial x_j}(\rho Y_i u_j) = \nabla \vec{J}_i + R_i, \tag{1}$$

where ρ is the flue gas mass density, Y_i concentration of the component, and u_j are the gas velocity components. The right-hand side terms describe the laminar and turbulent diffusion of the components and either production or consumption due to chemical reactions within the R_i term.

The above equations of component mass balance are accompanied by the equation of total mass balance

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u_j)}{\partial x_j} = 0. \tag{2}$$

Equations of momentum conservation are as follows

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu_{\text{eff}}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_l}{\partial x_l}\right)\right] + g_i, \tag{3}$$

where $\vec{g} = [g_1, g_2, g_3]$ is the external force acting on the fluid, in our case the gravity. The effective friction coefficient μ_{eff} is calculated from the turbulence model as

$$\mu_{\text{eff}} = \mu + \rho C_\mu \frac{k^2}{\epsilon},$$

where μ is the laminar viscosity, k the turbulent kinetic energy, and ϵ the turbulent energy dissipation rate. Constant C_μ, like additional constants mentioned later in the description of the turbulence model, has to be chosen empirically for the particular problem, in our case we use $C_\mu = 0.09$, which appears to give satisfactory results. All empirical constants in the turbulence model stated here are taken from [13].

The last equation describes the conservation of energy

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_j}(\rho u_j h) = -n_{\text{coal}}\frac{\mathrm{d}m_{\text{coal}}}{\mathrm{d}t}h_{\text{comb}} + q_r + q_c + q_s, \tag{4}$$

where the right-hand side terms are the heat of combustion, heat transfer by radiation, heat transfer by conduction, and heat source or sink. The heat transfer terms are computed as follows

$$-q_c = \nabla \cdot (\lambda \nabla T),$$

for the transfer by conduction, which is described by the Fourier law of head conduction, and

$$-q_r = \nabla \cdot \left(cT^3 \nabla T\right),$$

for the transfer by radiation. The radiation heat transfer is fully described by an integral-differential equation of radiation, which is very computationally expensive to solve. However, as the flue gas can be considered an optically thick matter, the above approximation of the radiation flux called Rosseland radiation model can be applied [13].

The heat sink term is nonzero only in the edge computation cells and describes the energy exchange with the walls of the furnace via conduction and radiation

$$q_s = A(T_{\text{gas}} - T_{\text{wall}}) + B(T_{\text{gas}}^4 - T_{\text{wall}}^4),$$

where A and B are constants dependent on the properties of the interface between the modeled region and its surroundings.

The particle mass change rate is currently described by one-step Arrhenian kinetics, which is used separately for the char and volatile coal components — combustion of the volatiles is more rapid than combustion of the char

$$\frac{\mathrm{d}m_p}{\mathrm{d}t} = -A_v m_p^\alpha [\mathrm{O}_2]^\beta \exp\left(-\frac{E_v}{RT_p}\right),$$

where m_p is the particle combustible mass, A_v, E_v are empirical constants, $[\mathrm{O}_2]$ oxygen concentration and T_p is the particle temperature.

These equations are accompanied by the equation of state, as usually

$$p = (\kappa - 1)\rho_{\text{gas}}\left(e_{\text{gas}} - \frac{1}{2}v_{\text{gas}}^2\right).$$

Here, κ is the Poisson constant and e_{gas} is the gas energy per unit mass.

For the turbulence modeling, we use the standard k-ϵ model, which describes the evolution of turbulence using two equations — first one for turbulent kinetic energy

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}(\rho k u_j) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + G_k - \rho\epsilon, \tag{5}$$

and the second one for turbulent kinetic energy dissipation rate

$$\frac{\partial}{\partial t}(\rho\epsilon) + \frac{\partial}{\partial x_j}(\rho\epsilon u_j) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\frac{\partial\epsilon}{\partial x_j}\right] + C_{1\epsilon}\frac{\epsilon}{k}G_k - C_{2\epsilon}\rho\frac{\epsilon^2}{k}. \tag{6}$$

Constants in this model have again to be determined empirically, in our case we use the following values: $C_{1\epsilon} = 1.44$, $C_{2\epsilon} = 1.92$, $\sigma_k = 1.0$, $\sigma_\epsilon = 1.3$.

Left hand sides of the equations describe passive advection of the respective quantities by the advection velocity $\vec{u}$. Right hand sides describe their spatial diffusion, their production and dissipation.

The term G_k, which describes the production of turbulence, can be derived from the Reynolds averaging process and written in the terms of the fluctuating part of the velocity as

$$G_k = \tau_{jl}\frac{\partial u_j}{\partial x_l} = -\rho\overline{u_j' u_l'}\frac{\partial u_j}{\partial x_l},$$

where τ_{jl} is the Reynolds stress tensor. However during practical computation, fluctuations u_j' and u_l' are unknown. Using the Boussinesq hypothesis, that the Reynolds stress is proportional to the mean strain rate

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right),$$

one can write turbulent production in a closed form

$$G_k = \mu_t S^2, \quad S = (2S_{jl}S_{jl})^{1/2}.$$

Diffusion of the species consists of two processes — laminar and turbulent, and the diffusion term in Eq. (1) can be written in the form

$$\vec{J}_i = -\left(\rho D_{i,m} + \frac{\mu_t}{\mathrm{Sc}_t}\right)\nabla Y_i.$$

First term corresponds to linear laminar diffusion, the second one to turbulent diffusion. Given the fact that the turbulent diffusion generally predominates the laminar, and the term $D_{i,m}$ is difficult to determine, the laminar diffusion can be usually ignored. The Sc_t coefficient is the turbulent Schmidt number and we put $\mathrm{Sc}_t = 0.7$.

To be able to model the particle phase, especially surface area of the particles, we still have to track the numerical density of the particles using the equation similar to the mass balance equation

$$\frac{\partial n_{\mathrm{coal}}}{\partial t} + \frac{\partial(n_{\mathrm{coal}}u_{\mathrm{coal}})}{\partial x_1} + \frac{\partial(n_{\mathrm{coal}}v_{\mathrm{coal}})}{\partial x_2} = 0. \tag{7}$$
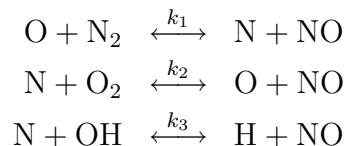
3   Simplified model of NOx chemistry

This model has been developed to approximately describe the amounts of NOx emissions leaving a coal combustion furnace. The real mechanism of coal flue gas production seems to be very complicated, so that just the most important phenomena and reaction paths

were considered to provide maximum possibility of using this model in real-time control and operation systems.

In most cases, NOx is interpreted as a group of NO and nitrogen dioxide (NO_2), which strongly pollute our living environment. There are two major processes attributing to the total NOx emissed. The former is known as *Thermal NOx* or *Zeldovich* and simply consists of oxidation of atmospheric nitrogen at high temperature conditions. The latter is called *Fuel NOx* and describes NOx creation from nitrogen, which is chemically bounded in coal fuel. Fuel NOx is usually the major source of NOx emissions. These are the only mechanisms involved, although a few more could be considered (such as *Prompt NOx (Fenimore)* or *Nitrous oxide (*N_2O*) intermediate* mechanisms).

3.1 Thermal NO

Thermal NO generation mechanism attributes only at high temperature conditions (approx. 1800 K) and is represented by a set of three equations, introduced by Zeldovich [5] and extended by Bowman [6]

$$
\begin{aligned}
O + N_2 \;\; &\xleftrightarrow{k_1} \;\; N + NO \\
N + O_2 \;\; &\xleftrightarrow{k_2} \;\; O + NO \\
N + OH \;\; &\xleftrightarrow{k_3} \;\; H + NO
\end{aligned}
$$

All these reactions are considered to be reversible. Rate constants were taken from [7] (see Tab. 1).

In order to compute the NO concentration, concentrations of nitrogen radical [N], oxygen radical [O] and hydroxyl radical [OH] must be known. It is useful to assume [N] to be in a quasi-steady state according to its nearly immediate conservation after creation. In fact, this N-radical formation is the rate limiting factor for thermal NO production, due to an extremely high activation energy of nitrogen molecule, which is caused by a triple bond between two nitrogen atoms. Hence, NO formation rate can be stated as

$$
\frac{d[NO]}{dt} \;=\; 2k_1^+ \cdot [O] \cdot [N_2] \cdot \frac{1 - \frac{k_1^- k_2^- [NO]^2}{k_1^+ [N_2] k_2^+ [O_2]}}{1 + \frac{k_1^- \cdot [NO]}{k_2^+ [O_2] + k_3^+ [OH]}}.
$$

Under certain conditions, oxygen molecule splits and recombines cyclically

$$
O_2 \;\; \xleftrightarrow{K_1} \;\; O + O,
$$
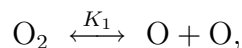
which can be profitably described by following partial equilibrium approach

$$
[O] \;=\; K_1 \cdot [O_2]^{1/2} \cdot T^{1/2}.
$$

As for OH radical, a similar partial equilibrium approach can be made, according to next reaction

$$
O + H_2O \;\; \xleftrightarrow{K_2} \;\; OH + OH
$$
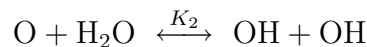
and the approach is

$$
[OH] \;=\; K_2 \cdot [O]^{1/2} \cdot [H_2O]^{1/2} \cdot T^{-0.57}.
$$

Equilibrium constants K_1 and K_2 are as follows

$$
\begin{aligned}
K_1 &= 36.64 \cdot \exp\left(\frac{-27123}{T}\right), \\
K_2 &= 2.129 \cdot 10^2 \cdot \exp\left(\frac{-4595}{T}\right).
\end{aligned}
$$

3.2 Fuel NO

Composition analysis show, that nitrogen-based species are more or less present in coal, usually in amounts of tenths to units of percent by weight. When the coal is heated, these species are transformed into certain intermediates and then into NO. Fuel itself is therefore a significant source of NO pollutants. When a coal particle is heated, it is presumed, that nitrogen compounds are distributed into volatiles and char. In many studies (e.g. [8]) it is unreasonably told, that half the nitrogen converts to volatiles and half into char. Since there is no reason for a presupposition like this, a parameter α is introduced to describe the distribution

$$
\begin{aligned}
m^{N}_{vol} &= \alpha \cdot m^{N}_{tot}, \\
m^{N}_{char} &= (1 - \alpha) \cdot m^{N}_{tot},
\end{aligned}
$$

where $\alpha \in <0,1>$, m^{N}_{tot} is the total mass of nitrogen, m^{N}_{vol} is the mass of nitrogen in volatiles and m^{N}_{char} is the mass of nitrogen in char.

As already mentioned, nitrogen transforms to pollutants via intermediates, which usually are ammonia NH_3 and hydrocyanide HCN. For further proceeding, a selection from four possible pathways must be made (see Fig. 1, ref. [9, 10]). To provide maximum complexity, another three parameters (similar to α) are introduced

- β is distribution of m^{N}_{tot} between HCN and NH_3 intermediates.

- γ is distribution of m^{N}_{HCN} between Pathway1 and Pathway2.

- δ is distribution of $m^{N}_{NH_3}$ between Pathway 3 and Pathway 4.

- $\beta,\ \gamma,\ \delta \in <0,1>$.

For example, mass of nitrogen in char entering Pathway 2 can be written as

$$
m^{N}_{P2,char} = m^{N}_{tot} \cdot \beta \cdot (1 - \gamma) \cdot (1 - \alpha).
$$

Different parametric studies should be carried out to find the best values of α, β, γ and δ suitable for specific type of coal. Five overall reactions of either NO formation or depletion were incorporated in the combustion part of the numerical code.

3.2.1 NO, HCN, NH$_3$ reactions

According to [11], formation rates of reactions

$$HCN + O_2 \xrightarrow{R_1} NO + \dots$$
$$NH_3 + O_2 \xrightarrow{R_2} NO + \dots$$
$$HCN + NO \xrightarrow{R_3} N_2 + \dots$$
$$NH_3 + NO \xrightarrow{R_4} N_2 + \dots$$
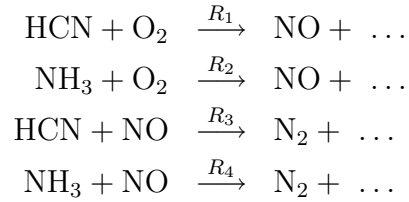
are given as

$$R_1 = 1.0 \cdot 10^{10} \cdot X_{HCN} \cdot X_{O_2}^a \cdot \exp\left(\frac{-33732.5}{T}\right),$$

$$R_2 = 4.0 \cdot 10^6 \cdot X_{NH_3} \cdot X_{O_2}^a \cdot \exp\left(\frac{-16111.0}{T}\right),$$

$$R_3 = -3.0 \cdot 10^{12} \cdot X_{HCN} \cdot X_{NO} \cdot \exp\left(\frac{-30208.2}{T}\right),$$

$$R_4 = -1.8 \cdot 10^8 \cdot X_{NH_3} \cdot X_{NO} \cdot \exp\left(\frac{-13593.7}{T}\right),$$

where X is the mole fraction and a is the oxygen reaction order taken from Tab. 2.

3.2.2 Heterogeneous NO reduction on char

Present char allows following adsorption process to occur

$$Char + NO \xrightarrow{R_5} N_2 + \dots$$

Levy [12] uses pore surface area (BET) to define NO source term

$$S_{ads}^{NO} = k_5 \cdot c_s \cdot A_{BET} \cdot M_{NO} \cdot p_{NO},$$

where $k_5 = 2.27 \cdot 10^{-3} \cdot \exp\left(\frac{-17168.33}{T}\right)$ is the rate constant, S_{ads}^{NO} is the NO source term, c_s is the concentration of particles, A_{BET} is the pore surface area and p_{NO} is the partial pressure of NO.

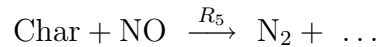   In order to evaluate overall NO source term, single source terms have to be summarized. This overall source term can be further used in transport equations. As for HCN and NH$_3$ source terms, it is possible to determine them from coal burnout rate. It is assumed, that nitrogen from both char and volatiles transforms to intermediate species quickly and totally.

4   Numerical algorithm

For numerical solution of the equations, finite volume method is used. For left and right hand sides in Eqs. (1), (2), (3), (4), (5), (6), (7), advection upstream splitting method (see [2]) is used to approximate fluxes in the FVM formulation, and edge dual-volume approximation is used to approximate the second order derivatives respectively. For detailed description of the solution procedure see [4].

5 Conclusion

We have developed a mathematical model, which approximates the combustion process in an industrial furnace, while being affordable from the computational complexity standpoint. As an outlook to the future, mainly the following improvement possibilities are being considered:

- Further refining of the coal combustion model.

- Evaluation and enhancements of the NOx generation model.

- Incorporation of the turbulent impact on the chemistry.

- Thorough evaluation of the 3D model.

Acknowledgments

Table 1: Rate constants for thermal NO chemical reactions, $k = A \cdot T^b \cdot \exp(-E_a/T)$.

Rate const.	A	b	E_a
k_1^+	$1.8 \cdot 10^8$	0	38370
k_2^+	$1.8 \cdot 10^4$	1	4680
k_3^+	$7.1 \cdot 10^7$	0	450
k_1^-	$3.8 \cdot 10^7$	0	425
k_2^-	$3.8 \cdot 10^3$	1	20820
k_3^-	$1.7 \cdot 10^8$	0	24560

Table 2: Oxygen reaction order.

Oxygen mole fraction	a
$X_{O_2} \leq 4.1 \cdot 10^{-3}$	1
$4.1 \cdot 10^{-3} \leq X_{O_2} \leq 1.11 \cdot 10^{-2}$	$-3.95 - 0.9 \cdot \ln X_{O_2}$
$1.11 \cdot 10^{-2} \leq X_{O_2} \leq 0.03$	$-0.35 - 0.1 \cdot \ln X_{O_2}$
$X_{O_2} \geq 0.03$	0
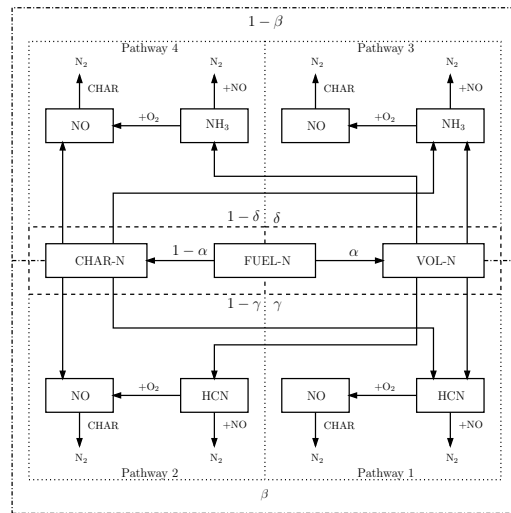
Figure 1: Fuel NO pathways.
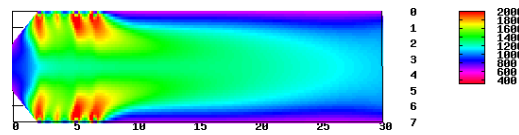
Figure 2: Temperature profile — symmetrical channel 30x7m, 2x4 burners, flue gas flow rate: 18 kg/s.
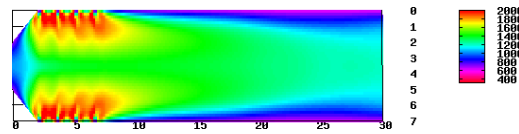
Figure 3: Temperature profile — flue gas flow rate: 28 kg/s.
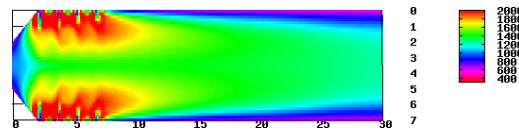
Figure 4: Temperature profile — flue gas flow rate: 38 kg/s.
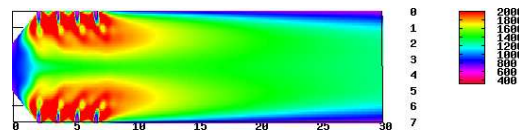
Figure 5: Temperature profile — flue gas flow rate: 48 kg/s.
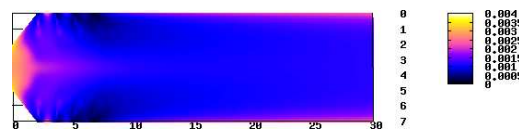
Figure 6: NOx profile — flue gas flow rate: 48 kg/s

References

[1] Y. C. Guo, C. K. Chan, *Fuel* 79 (12) (2000) 1467–1476.

[2] Meng-Sing Liou, C. Steffen, Jr., *J. Comp. Phys.* 107 (1) (1993) 23–29.

[3] J. Makovička, V. Havlena, in:M. Beneš, J. Mikyška, T. Oberhuber (Eds.), *Proceedings of the Czech-Japanese Seminar in Applied Mathematics 2004*, Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, 2005, p. 106.

[4] J. Makovička, V. Havlena, M. Beneš, in: A. Handlovičová, Z. Krivá, K. Mikula, D. Ševčovič (Eds.), *ALGORITMY 2002 Proceedings of contributed papers*, Publ. house of STU, 2002, p. 171.

[5] J. B. Zeldovich, *Acta Physicochimica* 21 (1946) 577–628.

[6] C. T. Bowman, D. J. Seery, *Emissions from Continuous Combustion Systems*, Plenum Press, New York, 1972, p. 123.

[7] NIST, *Chemical Kinetics Database on the Web*, National Institute of Standards and Technology, 2000, `http://www.kinetics.nist.gov`

[8] C. Kim, N. Lior, *Chemical Engineering Journal* 71 (3) (1998) 221–231.

[9] L. D. Smoot, P. J. Smith, *Coal Combustion and Gasification*, Plenum Press, New York, 1985, p. 373.

[10] F. C. Lockwood, C. A. Romo-Millares, *J. Inst. Energy* 65 (1992) 144–152.

[11] G. G. De Soete, *Proc. Combust. Inst.* 15 (1975) 1093–1102.

[12] J. M. Levy, L. K. Chen, A. F. Sarofim, J. M. Beer, *Proc. Combust. Inst.* 18 (1981) 111–120.

[13] FLUENT Inc., *FLUENT user's guide*, 2005.

[14] MPI Forum, *MPI: A Message-Passing Interface Standard*, `http://www.mpi-forum.org/docs/`

Spojování klasifikátorů

David Štefka

1. ročník PGS, email: `david.stefka@gmail.com`
Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská, ČVUT
školitel: Martin Holeňa, Ústav informatiky, Akademie věd ČR

Abstract. Combining classifiers has been given more and more importance in the recent years. The experiments show that a team of several different classifiers can perform better than any individual classifier in the team. This is why the combination of multiple classifiers can be used to improve the quality of classification. This arcticle tries to summarize and describe the commonly used approaches found in the literature and to give a basic overview over this field.

Abstrakt. Kombinování klasifikátorů může být úspěšnou metodou pro zvýšení kvality klasifikace. Experimenty ukazují, že tým několika různých klasifikátorů často poskytuje lepší výsledky než libovolný z jednotlivých klasifikátorů v tomto týmu. Proto je v posledních letech věnována spojování klasifikátorů značná pozornost. Tento článek si klade za cíl popsat a shrnout nejčastěji používané přístupy a také dát čtenáři základní přehled o této problematice.

1 Úvod

Kombinování klasifikátorů je efektivní metoda pro zvýšení kvality (tj. přesnosti, správnosti, citlivosti, separability atd.) klasifikace. Pokud řešíme klasifikační úlohu, můžeme vytvořit několik různých klasifikátorů, nechat je samostatně predikovat cílovou třídu a poté jistým způsobem jednotlivé výsledky zkombinovat. Tato kombinace může být taková, že vliv silných stránek jednotlivých klasifikátorů se zesiluje, zatímco nedostatky jsou utlumovány. Při takovéto kombinaci může výsledný klasifikátor predikovat lépe než libovolný samostatný klasifikátor v týmu. Takové přístupy můžeme v literatuře najít pod různými jmény, např. "kombinování klasifikátorů", "agregace klasifikátorů", "fúze klasifikátorů", "selekce klasifikátorů", "směs expertů", ... V zásadě však můžeme rozlišit dva základní přístupy. Jsou to:

- **selekce klasifikátorů**, při které používáme jisté rozhodovací pravidlo k určení, který klasifikátor použijeme pro predikci konkrétního vzoru; pouze tento "expertní" klasifikátor je tedy použit pro finální predikci

- **agregace(kombinování) klasifikátorů**, kdy k vytvoření finální predikce je použito více klasifikátorů v týmu

V tomto článku se budeme dále zabývat pouze agregací klasifikátorů. Speciálním případem agregace klasifikátorů je kombinování výsledků několika klasifikátorů stejného typu, které se liší pouze svými parametry, nebo používají různé trénovací množiny. Tyto přístupy jsou v literatuře obvykle nazývány souborové metody.

Článek je strukturován následovně: v sekci 2 formalizujeme termín klasifikátor a uvedeme některé modely, které mohou být použity pro kombinování klasifikátorů. Sekce 3

popisuje některé často používané metody pro vytváření souborů klasifikátorů a sekce 4 popisuje některá agregační pravidla. Sekce 5 poté uzavírá článek.

2 Klasifikace

Pod pojmem *klasifikace* rozumíme proces třídění objektů (neboli vzorů) z tzv. *příznakového prostoru* $\mathcal{X}$ do několika disjunktních množin, nazývaných *třídy*. Příkladem může být rozpoznávání rukou psaných číslic pomocí počítačového programu. Software (nebo matematické zobrazení), který realizuje tuto úlohu, se nazývá klasifikátor. Formálně můžeme klasifikátor definovat jako zobrazení:

$$\phi : \mathcal{X} \rightarrow \{1, \ldots, N\}, \tag{1}$$

kde $\mathcal{X} \subseteq \mathcal{R}^n$ je n-dimenzionální příznakový prostor a $\{1, \ldots, N\}$ je množina indexů jednotlivých tříd (odpovídající třídy značíme $C_1, \ldots, C_N$). Optimální, bezchybový klasifikátor by měl také splňovat následující podmínku:

$$\forall \mathbf{x} \in \mathcal{X} : \phi(\mathbf{x}) = i \iff x \in C_i \tag{2}$$

Konstrukce klasifikátoru splňujícího (2) však většinou není možná, takže tato podmínka je obvykle porušena. Místo toho se stanovují jisté požadavky na velikost trénovací, resp. testovací chyby. Pro naučení korektní klasifikace neznámých vzorů jsou používány vzory z tzv. trénovací množiny $\mathcal{T} \subseteq \mathcal{X}$. Ke získání odhadu kvality predikce se používají vzory z tzv. testovací množiny $\mathcal{S} \subseteq \mathcal{X}$. Množiny $\mathcal{S}, \mathcal{T}$ obsahují vzory, které jsou již korektně klasifikovány expertem v dané oblasti, a od klasifikátoru je očekávána jistá míra generalizace znalostí obsažených v těchto datech na neznámá data z $\mathcal{X}$. Chyba (testovací nebo trénovací, dle použité množiny) klasifikátoru může být definována jako počet korektně klasifikovaných vzorů dělený počtem všech vzorů.

K nejčastěji používaným druhům klasifikátorů patří např. umělé neuronové sítě, rozhodovací stromy, Bayesův klasifikátor, k-NN klasifikátor (k nejbližších sousedů), nebo SVM (support vector machines) klasifikátory. Všechny tyto metody mohou poskytnout kvalitní přístup ke klasifikaci, přesto je občas nutné kvalitu klasifikace dále zvýšit. Pokud se rozhodneme ke zlepšení klasifikace použít kombinování klasifikátorů, zjistíme, že klasifikátory definované pomocí (1) nám dávají příliš málo informace. Pro spojování klasifikátorů bychom využili informaci typu "Jaká je 'váha klasifikace' do jednotlivých tříd?". Aby to bylo možné, definujme klasifikátor jako zobrazení:

$$\phi : \mathcal{X} \rightarrow [0, 1]^N, \tag{3}$$

kde $\phi(\mathbf{x}) = (\mu_1(\mathbf{x}), \ldots, \mu_N(\mathbf{x}))$ je vektor 'vah klasifikace' $\mathbf{x}$ do každé třídy.

Tento typ klasifikátoru se obvykle nazývá '*measurement*' klasifikátor, zatímco (1) se většinou nazývá *ostrý* klasifikátor. Často se můžeme setkat i s jinými typy klasifikátorů, např. *pořadový* klasifikátor (viz [9]), jehož výstupem je setříděný seznam všech možných indexů tříd (tj. třída s nejvyšší vahou klasifikace je první, třída s druhou nejvyšší vahou druhá, atd., až třída s nejnižší vahou klasifikace je poslední), nebo *posibilistický* klasifikátor (viz např. [6]), který se liší od 'measurement' klasifikátoru tím, že $\mu_i(\mathbf{x}) \in [0, \infty)$.

Kdykoliv v tomto článku dále budeme mluvit o klasifikátoru, máme na mysli 'measurement' klasifikátor.

'Measurement' klasifikátor může být převeden na ostrý a naopak:

$$\phi_{ostry}(\mathbf{x}) = \arg\max\{\mu_i(\mathbf{x})|i = 1, \ldots, N\} \qquad (4)$$

$$\phi_{measurement}(\mathbf{x}) = (\delta(1, \phi(\mathbf{x})), \ldots, \delta(N, \phi(\mathbf{x}))),$$

kde $\delta(i, \phi(\mathbf{x})) = 1$ pokud $\phi(\mathbf{x}) = i$, 0 jinak.

Interpretace $\phi(\mathbf{x})$ závisí především na tom, jaký typ klasifikátoru je použit; možné interpretace mohou být:

- **posteriorní pravděpodobnost**, $\mu_i = P(\mathbf{x} \in C_i|\mathbf{x})$, např. pro Bayesův klasifikátor

- $\mu_i = $ stupeň **příslušnosti x** k fuzzy množině C_i

- $\mu_i = $ normalizovaná **vzdálenost** k nejbližšímu reprezentantu třídy i, pro 1-NN klasifikátor

- $\mu_i = $ **počet nejbližších reprezentantů** třídy i (dělený k), pro k-NN klasifikátor

- . . .

Někdy jsou na čísla μ_i kladeny dodatečné požadavky, např. $\sum_i \mu_i(\mathbf{x}) = 1$ pro pravděpodobnostní interpretaci $\phi(\mathbf{x})$. Tyto podmínky však nejsou důležité pro samotné spojování klasifikátorů, proto budeme požadovat pouze $\mu_i \in [0, 1]$. Vzhledem k tomu, že se snažíme o obecný popis přístupů spojování klasifikátorů, nezajímáme se také o interpretaci μ_i a chápeme tyho hodnoty pouze jako nějaké obecné váhy klasifikace do třídy C_i.

3 Souborové metody

Je zřejmé, že pokud chceme vytvořit kvalitní tým klasifikátorů, měly by všechny tyto klasifikátory dostatečně kvalitně predikovat. Trénovací a testovací chyby jednotlivých klasifikátorů však nejsou jediným kritériem ovlivňujícím kvalitu výsledného klasifikátoru – důležitá je také rozmanitost klasifikátorů. Tým navzájem nezávislých a rozmanitých klasifikátorů může dosahovat dobrých výsledků. Jsou-li však klasifikátory v týmu "pozitivně závislé", tj. mají závislé chyby (jestliže některý klasifikátor predikuje špatně, potom také mnoho ostatních klasifikátorů predikuje špatně), tým takovýchto klasifikátorů obvykle dosahuje horších výsledků než nejlepší z klasifikátorů v týmu. Klasifikátory v týmu mohou být však takové, že pouze několik málo klasifikátorů predikuje špatně a ostatní predikují správně pro každý vzor; v tomto případě nemá smysl tyto klasifikátory kombinovat, neboť predikci nelze dále zlepšovat.

K měření rozmanitosti týmu klasifikátorů můžeme použít různé míry – několik měr rozmanitosti je analyzováno a porovnáno mezi sebou v [7]. Zde je také diskutována role rozmanitosti týmu na chybovost finálního klasifikátoru. Další míra rozmanitosti, založená na entropii, je popsána v [8]. Vliv rozmanitosti týmu klasifikátorů na výslednou kvalitu klasifikace je stále předmětem výzkumu.

Bagging Vytváříme soubor klasifikátorů $\phi_1, \ldots, \phi_k$, trénovací množina je $\mathcal{T} = \{\mathbf{x}_i \in \mathcal{X} | i = 1, \ldots, m\}$:

1. Nastavíme $j = 1$.

2. Vzorkujeme m-krát s vracením z trénovací množiny $\mathcal{T}$ s použitím stejnoměrného rozdělení, tj. $P(\mathbf{x}_i) = 1/m \ \forall i$, čímž získáme novou trénovací množinu $\mathcal{T}_j$.

3. Vytvoříme klasifikátor ϕ_j s trénovací množinou $\mathcal{T}_j$.

4. Pokud $j < k$, zvýšíme j o 1 a vrátíme se ke kroku (2), jinak skončíme s výstupem $\phi_1, \ldots, \phi_k$.

<div align="center">Obrázek 1: Algoritmus bagging</div>

V literatuře můžeme najít různé metody pro vytváření týmů rozmanitých klasifikátorů. Mezi nimi jsou často používány *souborové metody*. Tyto metody konstruují množinu klasifikátorů stejného typu, které se liší pouze svými parametry nebo mají různé trénovací množiny. V tomto článku popíšeme metody nazývané bagging a boosting, které pracují s modifikacemi trénovací množiny. Setkáváme se však i s metodami kostruujícími např. k-NN klasifikátory s různými metrikami ([1]), SVM klasifikátory s různými jádrovými funkcemi, nebo neuronové sítě s různými architekturami.

Idea metody bagging (Bootstrap AGGregatING) byla publikována Breimanem v [2]. Bagging vybírá z trénovací množiny vzorky pomocí metody zvané bootstrap (náhodný, rovnoměrně rozdělený výběr s vracením, stejné velikosti jako je trénovací množina), a takto vzniklou množinu vzorů použije jako novou trénovací množinu pro konkrétní klasifikátor. Někreré vzory jsou vybrány více než jednou, některé naopak nejsou v nové trénovací množině přítomny vůbec. Algoritmus bagging je popsán na Obr. 1.

Jiný přístup, prezentovaný např. v [4], se nazývá boosting. Boosting je obecná metoda strojového učení pro zvýšení kvality slabého (tj. jednoduchého) klasifikátoru, která pracuje tak, že opakovaně pouští daný slabý algoritmus, přičemž trénovací data jsou vybírána v každém běhu na základě jiného rozdělení. V tomto článku popíšeme algoritmus AdaBoost (Freund and Schapire [4]). AdaBoost používá techniku podobnou bootstrapu, pouze s tím rozdílem, že pravděpodobnost výběru vzoru z trénovací množiny není pro všechny vzory stejná a v čase se mění. První klasifikátor je vytvořen pomocí bootstrapu (se stejnoměrným rozdělením). Poté spočteme trénovací chybu tohoto klasifikátoru (pomocí původní trénovací množiny) a těm vzorům, které jsou klasifikovány špatně, zvýšíme pravděpodobnost výběru. (Pokud je trénovací chyba klasifikátoru příliš vysoká, např. vyšší než 0.5 pro binární klasifikátor, tento klasifikátor nepoužijeme a pravděpodobnosti výběru jednotlivých vzorů nastavíme opět stejnoměrně.) Poté celý proces výběru opakujeme. Když máme zkonstruován požadovaný počet klasifikátorů, je agregace obvykle prováděna pomocí váženého průměru, kde jednotlivé váhy korespondují s příslušnými trénovacími chybami. Samozřejmě však můžeme použít libovolné jiné agregační pravidlo.

Zde popíšeme algoritmus AdaBoost.M1, který je použitelný pro binární klasifikaci. V [4] je popsána také modifikace pro klasifikaci do více tříd, algoritmus AdaBoost.M2. Tyto dvě metody splývají v případě binární klasifikace a liší se pouze v problémech klasifikace

AdaBoost.M1 Vytváříme soubor klasifikátorů $\phi_1, \ldots, \phi_k$, trénovací množina je $\mathcal{T} = \{\mathbf{x}_i \in \mathcal{X} | i = 1, \ldots, m\}$; pro každý trénovací vzor $\mathbf{x}_i$ známe odpovídající index třídy y_i:

1. Nastavíme $j = 1$.

2. Nastavíme diskrétní pravděpodobnosti $P_j(\mathbf{x}_i) = 1/m \; \forall i$.

3. Vzorkujeme m-krát s vracením z trénovací množiny $\mathcal{T}$, pomocí diskrétního rozdělení P_j, čímž získáme novou trénovací množinu $\mathcal{T}_j$.

4. Vytvoříme klasifikátor ϕ_j s trénovací množinou $\mathcal{T}_j$.

5. Spočítáme váženou chybu klasifikátoru ϕ_j:

$$err_j = \sum_{i:\phi(\mathbf{x}_i) \neq y_i} P_j(\mathbf{x}_i)$$

6. Pokud $err_j \geq 0.5$, odstraníme klasifikátor ϕ_j a vrátíme se k (2).

7. Nastavíme $\beta_j = err_j/(1 - err_j)$.

8. Aktualizujeme P:

$$P_{j+1}(\mathbf{x}_i) = \frac{P_j(\mathbf{x}_i)}{C} \cdot \begin{cases} \beta_j & pro \; \phi(\mathbf{x}_i) = y_i \\ 1 & jinak \end{cases},$$

kde $C = \sum_i P_j(\mathbf{x_i})$ je normalizační konstanta.

9. Pokud $j < k$, zvýšíme j o 1 a vrátíme se k (3), jinak skončíme s výstupem $\phi_1, \ldots, \phi_k$. Pokud je k agregaci klasifikátorů použit vážený průměr, nastavíme váhy $\omega_j = \log \frac{1}{\beta_j}$.

Obrázek 2: Algoritmus AdaBoost.M1

do tří a více tříd; z důvodů lepší srozumitelnosti popíšeme pouze AdaBoost.M1. Je popsán na Obr. 2.

Metody boosting a bagging jsou experimentálně porovnány v [4], kde ve většině případů dosahuje lepších výsledků algoritmus boosting. V [3] se autoři zabývají teoretickými aspekty baggingu a boostingu a je zde prezentována nová třída souborových metod, nazývaná arcing (speciálním případem této metody je boosting).

4 Agregace klasifikátorů

Poté co vytvoříme soubor skládající se z k klasifikátorů $\phi_1, \ldots, \phi_k$, je nutné jejich výsledky agregovat, abychom získali finální klasifikátor. Výstup souboru můžeme strukturovat do matice $k \times N$:

$$
\begin{pmatrix} \phi_1(\mathbf{x}) \\ \phi_2(\mathbf{x}) \\ \vdots \\ \phi_k(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \mu_{1,1}(\mathbf{x}) & \mu_{1,2}(\mathbf{x}) & \dots & \mu_{1,N}(\mathbf{x}) \\ \mu_{2,1}(\mathbf{x}) & \mu_{2,2}(\mathbf{x}) & \dots & \mu_{2,N}(\mathbf{x}) \\ & & \ddots & \\ \mu_{k,1}(\mathbf{x}) & \mu_{k,2}(\mathbf{x}) & \dots & \mu_{k,N}(\mathbf{x}) \end{pmatrix}, \tag{5}
$$

kde $\mu_{i,j}(\mathbf{x})$ je váha klasifikace $\mathbf{x} \in C_j$ přiřazená klasifikátorem ϕ_i. V [6] je tato matice nazývána rozhodovací profil.

Pro agregaci výstupů klasifikátorů použijeme agregační pravidlo $\mathcal{F}$:

$$
\Phi(\mathbf{x}) = \mathcal{F}(\phi_1(\mathbf{x}), \dots, \phi_k(\mathbf{x})), \tag{6}
$$

kde Φ je finální klasifikátor. Obvykle je Φ 'measurement' klasifikátor a pro finální predikci je jeho výstup převeden na ostrý použitím (4).

Velmi dobrý přehled a experimentální porovnání různých agregačních pravidel je možné nalézt v [6]. V této práci uvedeme některá v literatuře často se vyskytující agregační pravidla. Tato pravidla můžeme rozdělit do čtyř hlavních skupin — aritmetická pravidla, pravděpodobnostní pravidla, fuzzy pravidla a hierarchická klasifikace.

4.1 Aritmetická pravidla

Nejjednodušší pravidla používají pro agregaci výstupů klasifikátorů pouze jednoduché aritmetické operace. Můžeme použít např.:

- **hlasování** (pro ostré klasifikátory) – každý klasifikátor má jeden hlas a třída s maximálním počtem hlasů je prohlášena za výstup. Patové situace jsou řešeny libovolně.

- **maximum** – pro každý sloupec matice (5), použijeme pouze maximální hodnotu, tedy:
$$
\mu_j(\mathbf{x}) = \max\{\mu_{i,j}(\mathbf{x}) | i = 1, \dots, k\}, \ j = 1, \dots, N
$$

- **minimum** – stejné jako předchozí, pouze používáme minimální hodnotu:
$$
\mu_j(\mathbf{x}) = \min\{\mu_{i,j}(\mathbf{x}) | i = 1, \dots, k\}, \ j = 1, \dots, N
$$

- **průměr** – tento přístup se snaží aproximovat "typickou váhu klasifikace" pro každou třídu přes všechny klasifikátory:
$$
\mu_j(\mathbf{x}) = \frac{1}{k} \sum_{i=1}^{k} \mu_{i,j}(\mathbf{x}), \ j = 1, \dots, N
$$

- **vážený průměr** – předpokládejme, že máme definovánu váhu ω_i pro každý klasifikátor (např. váhy z algoritmu AdaBoost.M1). Pomocí těchto vah můžeme přiřadit větší důležitost těm klasifikátorům, které predikují lépe než ostatní:
$$
\mu_j(\mathbf{x}) = \frac{\sum_{i=1}^{k} \omega_i \mu_{i,j}(\mathbf{x})}{\sum_{i=1}^{k} \omega_i}, \ j = 1, \dots, N
$$

- **součin** – váha klasifikace do třídy C_i je dána jako součin hodnot i-tého sloupce matice (5):

$$\mu_j(\mathbf{x}) = \prod_{i=1}^{k} \mu_{i,j}(\mathbf{x}), \ j = 1, \ldots, N \tag{7}$$

4.2 Pravděpodobnostní pravidla

Zde uvedeme dvě různá agregační pravidla, která využívají teorie pravděpodobnosti – behavior knowledge space (BKS) a součinové pravidlo. BKS vytváří všechny možné kombinace indexů tříd a v každé kombinaci je vybrána nejpravděpodobnější třída. Součinové pravidlo aproximuje aposteriorní pravděpodobnost, že vzor náleží do j-té třídy pro dané j.

- **behavior knowledge space (BKS)** – tento přístup pracuje s ostrými klasifikátory. Nejprve vygenerujeme všechny možné kombinace ostrých výstupů (indexů tříd) a poté v každé kombinaci prohlásíme za výstup nejčastější třídu (mezi těmi vzory z trénovací množiny, pro které klasifikátory v souboru dávají odpovídající kombinaci indexů tříd). Například pokud máme $k = 5$ klasifikátorů a $N = 3$ třídy, možné kombinace indexů jsou:

$$[1,1,1,1,1][1,1,1,1,2][1,1,1,1,3][1,1,1,2,1] \ldots [3,3,3,3,3]$$

Pokud je v první kombinaci $[1,1,1,1,1]$ (tj. vzory trénovací množiny, pro které každý klasifikátor predikuje třídu C_1) r, s, t reprezentantů tříd C_1, C_2 a C_3, je za finální výsledek v této kombinaci brána třída $\arg\max_{C_1,C_2,C_3}\{r, s, t\}$. Patové situace jsou řešeny libovolně.

- **součinové pravidlo** – pokud můžeme výstupy klasifikátorů interpretovat jako odhady aposteriorních pravděpodobností, tj. $\mu_{i,j}(\mathbf{x}) = P_i(\mathbf{x} \in C_j|\mathbf{x})$, a klasifikátory $\phi_1, \ldots, \phi_k$ jsou navzájem nezávislé, pak má součin hodnot sloupců matice (5) (popsaný v předchozí podkapitole) také pravděpodobnostní interpretaci . Pokud vezmeme v úvahu také apriorní pravděpodobnosti jednotlivých tříd $P(C_1), \ldots, P(C_N)$, $P(C_j) = \frac{\#\{\mathbf{x} \in \mathcal{T} | \mathbf{x} \in C_j\}}{\#\mathcal{T}}$, dostaneme finální Bayesovsky optimální odhad:

$$\mu_j(\mathbf{x}) = \frac{\prod_{i=1}^{k} \mu_{i,j}(\mathbf{x})}{P(C_j)^{k-1}}, \ j = 1, \ldots, N$$

4.3 Fuzzy pravidla

Pokud můžeme interpretovat třídy $C_1, \ldots, C_N$ jako fuzzy množiny a výstupy klasifikátorů $\mu_i(\mathbf{x})$ jako hodnoty příslušnosti $\mathbf{x}$ do třídy C_i, můžeme pro agregaci klasifikátorů použít přístupy fuzzy logiky. Porovnání fuzzy a 'nonfuzzy' přístupů pro spojování klasifikátorů lze nalézt v [5].

- **fuzzy integrál** – nechť je univerzem množina všech klasifikátorů $\mathcal{U} = \{\phi_1, \ldots, \phi_k\}$, g fuzzy míra na $\mathcal{U}$ (reprezentující důležitost, nebo kvalitu klasifikátorů) a $A_j(\mathbf{x}) =$

Fuzzy integrál

1. Zvolíme k fuzzy hustot $g^{(1)}, \ldots, g^{(k)}$. Tyto hodnoty mohou být např. odhady přesností klasifikátorů.

2. Spočítáme λ jako jediný reálný kořen větší než -1 rovnice:

$$\lambda + 1 = \prod_{i=1}^{k}(1 + \lambda g^{(i)})$$

3. Kdykoliv je vzor $\mathbf{x}$ předložen ke klasifikaci, provedeme kroky (4)-(7) pro $j = 1, \ldots, N$.

4. Setřídíme j-tý sloupec matice (5) v rostoucím pořadí, čímž získáme hodnoty příslušnosti $\mu_{i_1,j}(\mathbf{x}), \ldots, \mu_{i_k,j}(\mathbf{x})$.

5. Setřídíme fuzzy hustoty v odpovídajícím pořadí: $g^{(i_1)}, \ldots, g^{(i_k)}$ a nastavíme $g(1) = g^{(i_1)}$.

6. Pro $t = 2, \ldots, k$ spočítáme

$$g(t) = g^{(i_t)} + g(t-1) + \lambda g^{(i_t)}g(t-1)$$

7. (a) Pro Sugenův integrál, finální hodnota je

$$\mu_j(\mathbf{x}) = \max_{t=1,\ldots,k}\{\min\{\mu_{i_t,j}(\mathbf{x}), g(t)\}\}$$

   (b) Pro Choquetův integrál, finální hodnota je

$$\mu_j(\mathbf{x}) = \mu_{i_1,j}(\mathbf{x}) + \sum_{t=2}^{k}\left(\mu_{i_{t-1},j}(\mathbf{x}) - \mu_{i_t,j}(\mathbf{x})\right)g(t-1)$$

Obrázek 3: Fuzzy integrál

$(\mu_{1,j}(\mathbf{x}), \ldots, \mu_{k,j}(\mathbf{x}))^T$ j-tý sloupec matice (5) (reprezentující příslušnost $\mathbf{x}$ do třídy C_j pro všechny klasifikátory). $A_j(\mathbf{x})$ může být chápána jako fuzzy množina na $\mathcal{U}$. Fuzzy míra g může být spočtena z k bodových hodnot míry $g^{(1)}, \ldots, g^{(k)}$ (nazývané fuzzy hustoty). Poté můžeme použít Sugenův nebo Choquetův fuzzy integrál pro agregaci $A_j(\mathbf{x})$ pomocí g. Agregace pomocí fuzzy integrálu (jak je popsána v [5]) je zobrazena na Obr. 3.

- **rozhodovací šablony** – tato technika je popsána v [6]. Pro každou třídu C_j je definována rozhodovací šablona, která vyjadřuje typický výstup souboru pro tuto třídu, jako $k \times N$ matice DT_j, jejíž prvky jsou průměry výstupů souboru přes všechny trénovací vzory náležící do C_j:

$$(DT_j)_{r,s} = \frac{\sum_{\mathbf{x}\in\mathcal{T}:\mathbf{x}\in C_j}\mu_{r,s}(\mathbf{x})}{\#\{\mathbf{x}\in\mathcal{T}:\mathbf{x}\in C_j\}} \tag{8}$$

Matice (5) a DT_j interpretujeme jako fuzzy množiny na univerzu $\mathcal{U}$ s $k \times N$ prvky. Pokud je ke klasifikaci dán vzor $\mathbf{x}$, spočítáme výstup souboru (5) a spočítáme nějakou míru podobnosti tohoto výstupu a rozhodovací šablony pro každou třídu. Tyto míry podobnosti mohou být např. fuzzy míry podobnosti těchto množin, míry fuzzy inkluze, míry konzistence, nebo libovolná maticová norma rozdílu těchto matic. V [6] je popsáno 11 různých měr podobnosti a také je zde experimentálně porovnán efekt jednotlivých měr. Podle experimentálních výsledků prací [6] a [5] můžeme doporučit Euklidovskou normu, nebo S_3 míru podobnosti: nechť A, B jsou fuzzy množiny na $\mathcal{U}$ a μ_A, μ_B jejich funkce příslušnosti. Potom definujme

$$S_3(A, B) = 1 - ||A \triangle B||,$$

kde $||A||$ značí relativní kardinalitu fuzzy množiny A a

$$A \triangle B = (A \cap \bar{B}) \cup (\bar{A} \cap B),$$

kde $\cap$ je fuzzy průnik, $\cup$ fuzzy sjednocení a $\bar{\cdot}$ je fuzzy komplement.

4.4 Hierarchická klasifikace

Na výstup (5) souboru klasifikátorů se můžeme dívat také jako na vektor příznaků $\in [0, 1]$ délky kN. Tento prostor můžeme použít jako vstup pro jiný ostrý klasifikátor "vyšší úrovně" $\Phi : [0, 1]^{kN} \rightarrow \{1, \ldots, N\}$. Tento přístup má nevýhodu v tom, že pokud klasifikátory $\phi_1, \ldots, \phi_k$ predikují "skoro ostré indexy" (tj. pouze jedna složka vektoru $\phi(\mathbf{x})$ je blízká jedné a ostatní složky jsou blízké nule), pak kovarianční matice trénovacích vzorů jsou blízké singulární matici. Proto kdykoliv používáme normální rozdělení pro modelování rozdělení vzorů ve třídách (např. u Bayesova klasifikátoru), výsledky mohou být nepřesné.

5 Závěr

V tomto článku byly popsány různé přístupy spojování klasifikátorů, byly uvedeny možné matematické modely klasifikátorů, které mohou být použity ke spojování klasifikátorů a také byly popsány dvě metody (bagging a boosting) pro vytváření souborů klasifikátorů. Také byly popsány některé často používané metody pro agregaci klasifikátorů. V budoucnu bych se chtěl zabývat fuzzy agregačními pravidly a zejména pro agregaci použít jiné typy fuzzy integrálů než Sugenův a Choquetův. Také mám v úmyslu zabývat se použitím Mamdani-Assilianova fuzzy regulátoru pro agregování klasifikátorů.

Literatura

[1] Y. Bao, N. Ishii, and X. Du. Combining multiple k-nearest neighbor classifiers using different distance functions. In 'IDEAL', 634–641, (2004).

[2] L. Breiman. *Bagging predictors.* Machine Learning **24** (1996), 123–140.

[3] L. Breiman. Bias, variance, and arcing classifiers. Technical report, Technical Report 460, Statistics Department, University of California, (1996).

[4] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In 'International Conference on Machine Learning', 148–156, (1996).

[5] L. I. Kuncheva. *Fuzzy versus nonfuzzy in combining classifiers designed by boosting.* IEEE Transactions on Fuzzy Systems **11** (2003), 729–741.

[6] L. I. Kuncheva, J. C. Bezdek, and R. P. W. Duin. *Decision templates for multiple classifier fusion: an experimental comparison.* Pattern Recognition **34** (2001), 299–314.

[7] L. I. Kuncheva and C. J. Whitaker. *Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy.* Machine Learning **51** (2003), 181–207.

[8] W. Liu, Z. Wu, and G. Pan. An entropy-based diversity measure for classifier combining and its application to face classifier ensemble thinning. In 'Sinbiometrics', 118–124, (2004).

[9] O. Melnik, Y. Vardi, and C.-H. Zhang. *Mixed group ranks: Preference and confidence in classifier combination.* IEEE Transactions on Pattern Analysis and Machine Intelligence **26** (2004), 973–981.

Solar Wind Electrons and their Anisotropy

Štěpán Štverák

3rd year of PGS, email: `stepan.stverak@centrum.cz`
Department of Mathematics, Faculty of Nuclear Science and Physical
Engineering, CTU
advisor: Pavel Trávníček*, Department of the Space Physics, Institute of
Atmospheric Physics, ASCR

Abstract. We present a statistical study of solar wind electrons using data from missions HELIOS I collected in the ecliptic plane covering the radial distance from the Sun from 0.3 up to 1 AU. We focused on the electron temperature anisotropy which control mechanisms are still not well understood. For this purpose we looked at the temperature anisotropy as a function of two important parameters, namely the electron collisional age A_e defined as a number of collision suffered by an electron during the expansion of the solar wind and the electron parallel plasma beta $\beta_{||}$, to see whether the electrons are constrained by some instabilities or driven by collisions. The temperature anisotropy was computed by fitting the measured electron velocity distribution functions (eVDF) with a core-halo model defined as a sum of a bi-Maxwellian and a bi-Kappa function representing the core and halo population respectively.

Abstrakt. Uvádíme statistickou studii vlastností elektronů ve slunečním větru na datech z družice HELIOS I naměřených v rovině ekliptiky ve vzdálenosti mezi 0.3 a 1 AU od Slunce. Zaměřili jsme se na teplotní anisotrpii, jejíž kontrolní mechanismy nejsou stále do detailu prozkoumány. Teplotní anisotropie je zde zkoumána jako funkce dvou důležitých parametrů, a sice průměrného počtu srážek uskutečněných během expanze slunečního větru a elektronového paralelního plasma beta, abychom mohli porovnat, do jaké míry je ovlivněna srážkami nebo možnými nestabilitami. Výpočet teplotní anisotropie byl proveden fitováním naměřených elektronových distribučních funkcí teoretickým modelem skládajícím se ze sumy bi-Maxwellovského a bi-Kappa rozdělení.

1 Introduction

Observed electron velocity distribution functions (eVDFs) of the solar wind permanently exhibit three different components: a thermal *core* and a supra-thermal *halo*, which are always present at all pitch angles, and a sharply magnetic field aligned *strahl* which is usually antisunward-moving [1]. At 1 AU, the core has a typical temperature of 10^5 K and represents about 95% of the total electron number density. The halo population has a typical temperature of 7×10^5 K and represents with the strahl the remaining portion of the total electron number density. The distribution functions of the main two populations, the core and halo, are isotropic in the plane perpendicular to the ambient magnetic field while they can differ in the parallel direction. We can thus speak about the parallel ($T_{||}$) and perpendicular ($T_\perp$) electron temperature which are defined by the second order moment of the eVDF - the pressure tensor.

---

*Special thanks also to my second advisor Milan Maksimovic (LESIA, OBSPM, Meudon, France) which is responsible for my jointly-supervised PGS in France.

The ratio between the parallel and perpendicular electron temperatures $T_{||}/T_{\perp}$ is typically found at 1 AU around 1.1 [1], thus the electron distributions are nearly isotropic while at their origin (closer to the Sun) the parallel temperature is usually greater then the perpendicular one. One of the basic issues of the solar wind plasma physics, which is still note well understood, is the question of what controls this isotropisation process during the solar wind expansion. The main two candidates which are believed to be responsible for this phenomenon are the Coulomb collision and the kinetic micro-instabilities related with the anisotropic particle distributions.

In this paper we present a statistical description of these effects on a sufficiently large dataset. We used eVDF measured onboard the HELIOS I spacecraft and we fitted them with an analytical model of the core and halo populations to get the estimations of the both temperatures and compared them with electron parallel plasma beta, which is related to the instabilities, and the electron collisional age, the number of collision suffered by an electron during the expansion of the solar wind.

2   Instrument and Data

For our data set we have used eVDFs measured onboard the spacecrafts HELIOS I. This data set includes roughly 100 000 measurements covering the radial distance in the ecliptic plane from 0.3 up to 1 AU. The time interval of these data samples starts from the beginning of the year 1975 to the end of 1978. The Helios I spacecraft [9] was launched in 1974 into an ecliptic orbit with a perihelion around 0.3 AU and aphelion reaching the Earth's orbit. The electrons onboard this satellite were measured by the energy analyzer I2 described by [11]. Basically the instrument measured 2-D distribution function with eight angular 15° wide sectors spaced 45° apart from one another. The range of the instrument covers energies from 0.5 to 1660 eV and the time resolution of the measurements is 40 s. Since we are interested in the $T_{||}/T_{\perp}$ ratio we need to have the distribution as a function of $v_{\perp}$ and $v_{||}$ where the indices have the same meaning as for the temperatures. Because the field of view of the detector lies in the ecliptic plane whatever is the direction of the magnetic field, we had to first make a preselection of all the measurements. In order to have eVDFs in the $(v_{\perp}, v_{||})$ plane only data samples where the angle between the magnetic field vector and the ecliptic plane was smaller then $\approx 6°$ were used for the analysis.

3   Method

The electrons in the solar wind typically exhibit three different populations: a nearly isotropic thermal *core*, representing roughly 95% of the total density, a suprathermal *halo* and a magnetic field aligned *strahl*. The strahl component isn't still well analytically described and it is thus hard to separate or fit it. Account on this, we use only the core-halo model

$$f_M = f_c + f_h \tag{1}$$

where f_c and f_h corresponds to the core and halo component respectively, and we fit only data points measured in the sunward direction where the strahl is not presented

(assuming that the core and halo populations have the same properties in the anti-sunward direction). We have also thrown away data points below the one count level. The core population is well described with the bi-Maxwellian

$$f_c = n_c \left(\frac{m}{2\pi k} \right)^{3/2} \frac{1}{T_{c\perp} \sqrt{T_{c\|}}} \exp \left[-\frac{m}{2k} \left(\frac{1}{T_{c\perp}} v_\perp^2 + \frac{1}{T_{c\|}} v_\|^2 \right) \right] \tag{2}$$

where n_c is the core density, m is the electron mass, k is the Boltzmann constant and $T_{c\perp}$ and $T_{c\|}$ are the core perpendicular and parallel temperatures respectively. The halo usually exhibits non-thermal properties and is therefore more convenient to fit it with a bi-Kappa distribution function

$$f_h = n_h \left(\frac{m}{\pi k(2\kappa - 3)} \right)^{\frac{3}{2}} \frac{1}{T_{h\perp} \sqrt{T_{h\|}}} \frac{\Gamma(\kappa + s)}{\Gamma(\kappa + s - \frac{3}{2})} \left(1 + \frac{m}{k(2\kappa - 3)} \left(\frac{1}{T_{h\perp}} v_\perp^2 + \frac{1}{T_{h\|}} v_\|^2 \right) \right)^{-\kappa - s} \tag{3}$$

where again n_h is the halo density, m is the electron mass, k is the Boltzmann constant and $T_{c\perp}$ and $T_{c\|}$ are the halo perpendicular and parallel temperatures respectively (in fact these are not exactly the temperatures but the corresponding moments of the eVDF). Maksimovic et al. [7] showed that this model is more convenient then the classical one – the sum of two bi-Maxwellians [1]. In total we have seven parameters to fit – $n_c, n_h, T_{c\perp}, T_{c\|}, T_{h\perp}, T_{c\|}$ and κ. For the fitting it's better to transform this set of parameters into another one – $n, \alpha, T, \beta, \gamma, \delta$ and κ - where the transform relations are as follows

$$\begin{aligned} n = n_c + n_h \quad & \alpha = \frac{n_h}{n_c + n_h} \quad && T = T_{c\|} \\ \beta = T_{c\|}/T_{c\perp} \quad & \gamma = T_{h\|}/T_{h\perp} \quad && \delta = T_{h\|}/T_{c\|} \end{aligned} \tag{4}$$

Using these substitutions we can better control the fitting process, mainly the anisotropy and the relation between the two different populations. The core and halo formulas written in the new variables are

$$f_c = n(1 - \alpha) \left(\frac{m}{2\pi k} \right)^{3/2} \beta \left(\frac{1}{T} \right)^{3/2} \exp \left[-\frac{m}{2k} \frac{1}{T} \left(\beta v_\perp^2 + v_\|^2 \right) \right] \tag{5}$$

and

$$f_h = n\alpha \left(\frac{m}{\pi k(2\kappa - 3)} \right)^{3/2} \gamma \left(\frac{1}{\delta T} \right)^{3/2} \frac{\Gamma(\kappa + s)}{\Gamma(\kappa + s - \frac{3}{2})} \left(1 + \frac{m}{k(2\kappa - 3)} \frac{1}{\delta T} \left(\gamma v_\perp^2 + v_\|^2 \right) \right)^{-\kappa - s} \tag{6}$$

Instead of f_M, we are fitting the logarithm of the distribution function, $\log f_M$, in order to take into account also the high energy tales of the halo population which are some orders of magnitude lower compared to the core part of the eVDF. Since the model is not linear in all of the parameters, we use an iterative fitting technique based on the well known Levenberg-Marquardt algorithm [8].

Typically the measured eVDFs are influenced by several effects and some corrections must be done before applying the fitting process. First of all the measured eVDFs are modified by the S/C potential by which the solar wind electrons are de/accelerated and

their energy is changed and therefore the computed moments of the eVDFs are incorrect (see [12]). However if we assume that the S/C potential is isotropic around the detector, the ratio $T_\perp/T_\parallel$ will remain unaffected. In this study, we didn't consider this effect and for further analysis we took the correct electron density from other electron/ion measurements. Second the core population is polluted by the cold photoelectrons emitted from the spacecraft body. This part of the eVDF has to be removed from the analysis using an energy cutoff. This energy threshold was set to 10 eV in our case. Last thing we have to do before the fitting is the velocity shift to the solar wind plasma frame. The eVDFs are measured in the spacecraft frame while our model expects plasma at rest thus the S/C velocity (usually negligible for the electrons) and the solar wind bulk speed, taken from the ion measurements, has to be subtracted from the measured one.

4   Results

The purpose of the present study is to statistically describe the temperature (an)isotropy of the solar wind electrons with respect to the electron collisions and the electron parallel plasma beta. We computed the temperature anisotropy by fitting the measured electron velocity distribution functions as described above. This method was applied on about 100 000 samples of measured eVDFs representing four consecutive years in the solar wind which can be taken as a sufficient statistical data set. For our iterative fitting algorithm we set the initial parameters to

$$n = 5.0e6 \quad \alpha = 0.05 \quad T = 1.0e5 \quad \kappa = 2.5$$
$$\beta = 1.0 \quad\;\; \gamma = 1.0 \quad\;\; \delta = 6.0$$

which are typical values for the solar wind and constrained them as

$$n \in (1.0e4, 5.0e8) \quad \alpha \in (0.001, 0.15) \quad T \in (1.0e3, 1.0e7) \quad \kappa \in (1.9, 10.0)$$
$$\beta \in (0.2, 8.0) \quad\quad\;\; \gamma \in (0.2, 8.0) \quad\quad\;\; \delta \in (3.0, 12.0)$$

Figure 1 shows one example of cuts trough a fitted eVDF representing all eight angular sectors measured with the I2 instrument. Only points represented by asterisks are fitted while points represent by dots are data which are not taken into account because of the strahl population or the one count level.

4.1   Instabilities

The typical conditions in the solar wind implies that the electrons are collisionless. It means that their collective behavior is driven by the wave-particle interactions, rather than by particle Coulomb collisions. For a sufficiently large anisotropy in the particle distribution function an plasma instability can arise which produces field fluctuations. They in turn lead to wave-particle interactions pushing the plasma again toward the isotropic state. The minimum anisotropy needed to give rise to these fluctuations, called the instability threshold, represents thus a constraint on the $T_\perp/T_\parallel$ ratios which can in the solar wind occur. There are in fact two cases of these instabilities.

The first one is the case where $T_\perp/T_\parallel > 1$. For a bi-Maxwellian distribution and a sufficiently homogeneous plasma, the fastest growing instability caused by this anisotropy
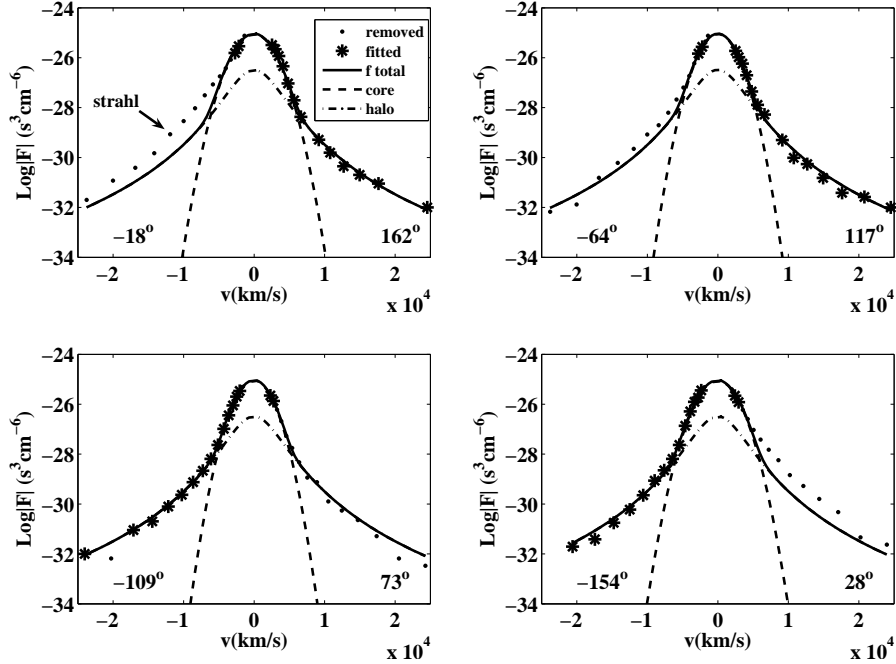
Figure 1: Example of fitted eVDF from Helios measurements. The panels show cuts trough the eVDF, the corresponding pitch angles are in the lower corners of the panels. Points represented by asterisks are fitted while points represent by dots are data which are not taken into account because of the strahl population or the one count level.

is the *whistler instability* [5]. The electrons are cyclotron resonant in this case while the protons are not, thus the proton temperature has no effect on the properties of this instability. In the opposite case where $T_\perp/T_{||} < 1$ the *electron firehose instability* can arise [3]. Here the electrons are nonresonant while now the protons are so the maximum growth rate for this instability is a function of the proton temperature. The curves of the constant maximum growth rates derived from the linear theory (neglecting the non-linear effects) can be approximated as

$$\frac{T_\perp}{T_{||}} = 1 + \frac{a}{\beta_{||}^b} \tag{7}$$

where the parameter a is positive for the whistler instability and negative for the firehose. The value $\beta_{||}$ is the electron parallel plasma beta defined as

$$\beta_{||} = \frac{8\pi n k T_{||}}{B^2} \tag{8}$$

where n is the electron density, k is the Boltzmann constant, $T_{||}$ is their parallel temperature and **B** is the background magnetic field. The β parameters measure the relative importance of the particle kinetic and magnetic field pressures. The plasma is called low-beta for $\beta < 1$ and high-beta for $\beta \gtrsim 1$, both cases are to be found in the solar wind plasma.

The results for our data set are displayed on fig. 2 as histograms of the relative frequencies of $T_\perp/T_{||}$ vs. $\beta_{||}$. The histograms are over plotted with curves representing the maximum growth rates for the whistler (dash-and-dot line) and the firehose (dashed
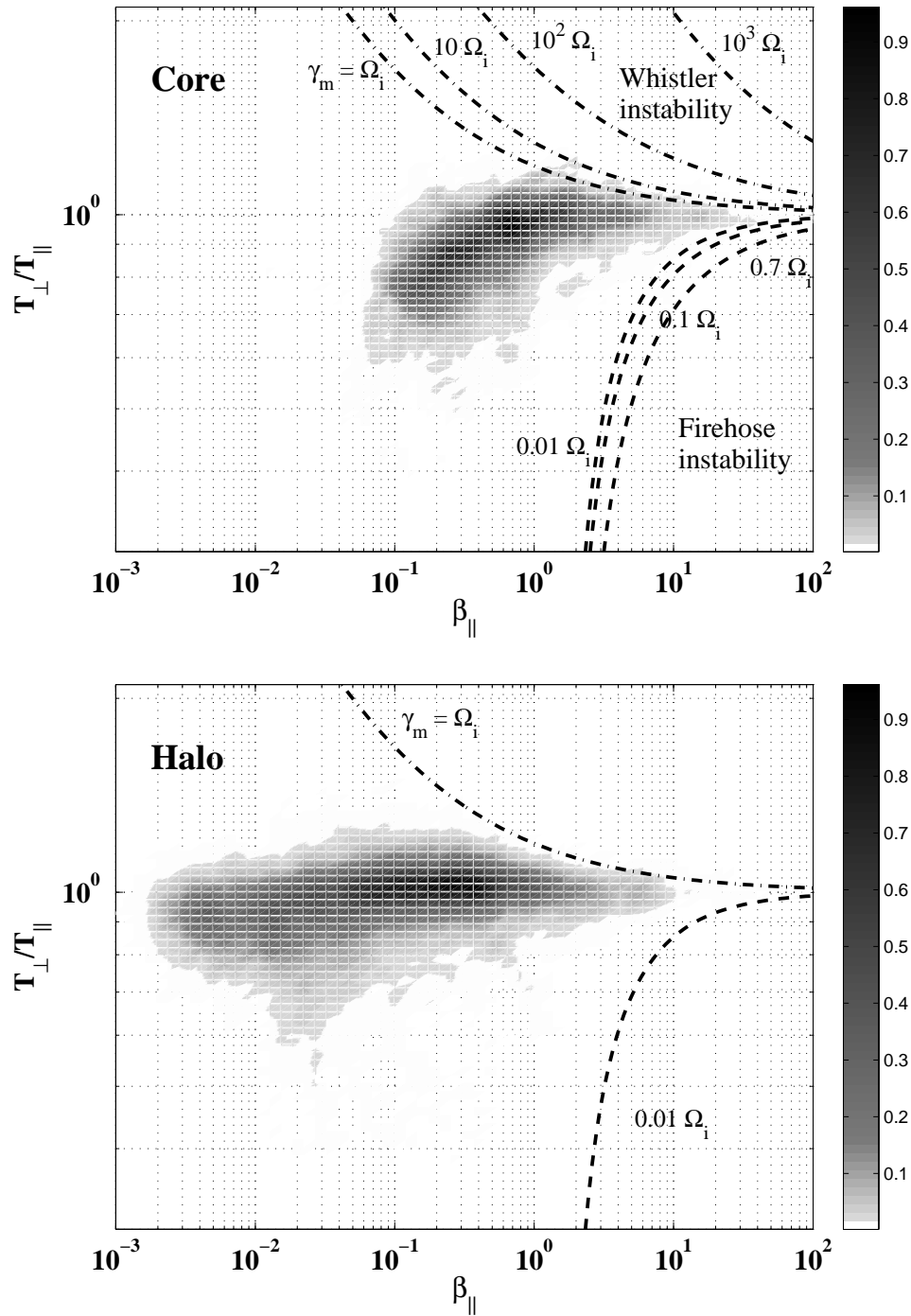
Figure 2: Relative frequency histograms of the $T_\perp/T_\parallel$ vs. $\beta_\parallel$ for the core (upper panel) and halo (lower panel) electron populations. The curves representing the maximum growth rates for the whistler (dash-and-dot line) and the firehose (dashed line) instability.

line) instability. While there is a nice correlation for the core (and also for the halo) with the whistler instability, it seems that for $T_\perp/T_\parallel < 1$ the anisotropy isn't completely controlled by the firehose instability.

4.2 Collisions

The solar wind plasma is usually treated as a collisionless medium because of its relatively low density and high temperature. With a typical density and temperature of 5 cm^{-3} and 1.5 K respectively at 1 AU the Coulomb mean free path length λ_m is of order 10^8 km which is comparable with the length scales of the system. Nevertheless it's interesting to see if the rare collisions can play any role in the isotropisation of the electrons in the solar wind.

In order to compute the number of collision suffered by an electron during the expansion of the solar wind from the corona, we can use the same way as it was done in [10]. The collision frequency between two electrons producing a transverse diffusion can be approximated as

$$\nu_{e\perp} \approx 7.710e^{-6} n_e T_e^{-3/2} \ln \Lambda \tag{9}$$

where T_e is the electron temperature in eV, n_e is the electron density given in cm^{-3}, and $\ln \Lambda$ ($\simeq 25.5$) is the Coulomb logarithm. If we add also the collisions with protons and α-particles, the total collision frequency will be

$$\nu_{e\perp,total} \approx 2.55\nu_{e\perp}. \tag{10}$$

Then the number of collisions - the collisional age A_e - is obtained by integrating the total frequency from some initial distance r_0 to distance R where the plasma parameters are measured. Thus

$$A_e = \int_0^{t_R} \nu_{e\perp,total}(\tau) \mathrm{d}\tau \tag{11}$$

where (v_{sw} denotes the solar wind bulk speed)

$$r = r_0 + tv_{sw} \quad \Rightarrow \quad t_R = \frac{R - r_0}{v_{sw}}. \tag{12}$$

and r_0 is the initial distance from which we count the collisions. If we assume a constant v_{sw} and that n_e and T_e vary with the radial distance as r^{-2} and $r^{-\alpha}$ respectively we can write

$$n_e(r) = \frac{K}{r^2} = \frac{KR^2}{R^2 r^2} = n_e(R)\frac{R^2}{r^2} \tag{13}$$

and similarly

$$T_e(r) = \frac{K}{r^\alpha} = \frac{KR^\alpha}{R^\alpha r^\alpha} = T_e(R)\frac{R^\alpha}{r^\alpha}. \tag{14}$$

Inserting this into (9) and using (12), the frequency as a function of time is

$$\nu_{e\perp,total} = 2.55\nu_{e\perp}(R)\left(\frac{R}{r_0 + tv_{sw}}\right)^{2-1.5\alpha} \tag{15}$$

where

$$\nu_{e\perp}(R) \approx 7.710e^{-6} n_e(R) T_e(R)^{-3/2} \ln \Lambda \tag{16}$$

Now integrating of (11) results in

$$A_e = 2.55\nu_{e\perp}(R)\frac{R}{v_{sw}}\left(\frac{1 - \left(\frac{R}{r_0}\right)^{1-1.5\alpha}}{1.5\alpha - 1}\right) \qquad (17)$$

The parameter α depends on the solar wind properties, it varies for the slow and fast wind and also for the core and halo population (see [4, 6, 2]). The values we have used are displayed in Tab. 1. The initial distance r_0 was set to 0.2 AU, less then the minimum distance of our data samples.

	core	halo
slow wind	0.4	
fast wind	0.6	0.3

Table 1: The variation exponent of the electron temperature α differs for the slow and the fast solar wind and depends also on the electron population (core/halo). For this study we have used values displayed in this table.

The correlation between the $T_\perp/T_\parallel$ ratio and the computed collisional age are represented on figure 3 in the similar manner as for the instabilities. The upper panel is again for the core population and the lower one shows the results for halo.

5 Discussion

Our resulting plasma parameters are in agreement with those usually observed in the solar wind. As we have shown both, the anisotropy instabilities and the electron collisions, constrain the $T_\perp/T_\parallel$ ratio in the solar wind plasma. In the case of the instabilities the upper limit of the temperature ratio is well corresponding with increasing plasma beta to the whistler electron instability for both the core and the halo population. The constraint by the firehose instability for $T_\perp/T_\parallel < 1$ is not so obvious. For the halo, it can be because of the used model where the electrons are supposed to have a bi-Maxwellian distribution while the real observations are closer to the bi-Kappa function. But also for the core there can be some other effects which bounds the ratio of the temperatures. Also the electron-particle Coulomb collisions, which are usually neglected because of their low frequency in the solar wind plasmas, are related with process of isotropisation of the velocity distribution function as displayed on figure 3 where we can see a small trend between the electron collisional age and the $T_\perp/T_\parallel$. Next step in our study will be to look at the evolution of the electron anisotropy with the increasing radial distance from the Sun including also data from other mission, namely the CLUSTER and ULYSSES, to cover the radial distance up to 4 AU.
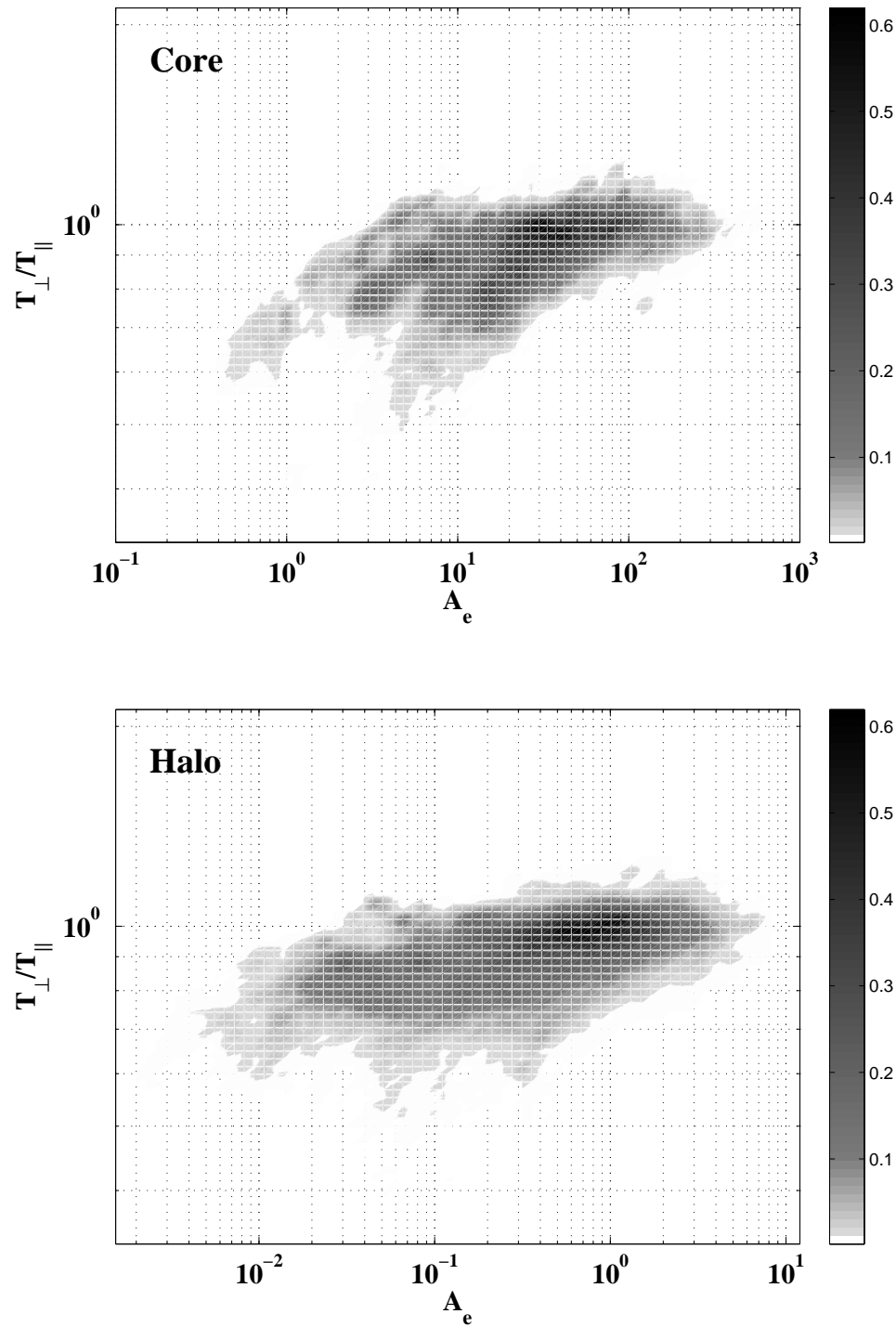
Figure 3: Relative frequency histograms of the $T_\perp/T_\parallel$ vs. A_e for the core (upper panel) and halo (lower panel) electron populations. A small trend can be seen for both populations pushing the temperature ration toward 1 with increasing number of collisions.

References

[1] W. Feldman, J. R. Asbridge, S. J. Bame, M. D. Montgomery, and S. P. Gary. *Solar wind electrons.* J. Geophys. Res. **80** (1975), 4181–4196.

[2] A. Fludra, G. Del Zanna, D. Alexander, and B. J. I. Bromage. *Electron density and temperature of the lower solar corona.* J. Geophys. Res. **104** (May 1999), 9709–9720.

[3] J. V. Hollweg and H. J. Volk. *New plasma instabilities in the solar wind.* J. Geophys. Res. **75** (1970), 5297.

[4] K. Issautier, N. Meyer-Vernet, M. Moncuquet, and S. Hoang. *Solar wind radial and latitudinal structure - electron density and core temperature from ulysses thermal noise spectroscopy.* J. Geophys. Res. **103** (February 1998), 1969.

[5] C. F. Kennel and H. E. Petscheck. *Limit on stably trapped particle fluxes.* J. Geophys. Res. **71** (1966), 1–28.

[6] M. Maksimovic, S. P. Gary, and R. M. Skoug. *Solar wind electron suprathermal strength and temperature gradients: Ulysses observations.* J. Geophys. Res. **105** (August 2000), 18337.

[7] M. Maksimovic, I. Zouganelis, J.-Y. Chaufray, K. Issautier, E. E. Scime, J. E. Littleton, E. Marsch, D. J. McComas, C. Salem, R. P. Lin, and H. Elliott. *Radial evolution of the electron distribution functions in the fast solar wind between 0.3 and 1.5 au.* J. Geophys. Res. **110** (2005).

[8] D. Marquardt. *An algorithm of least-squares estimation of nonlinear parameters.* J. Appl. Math. **11** (1963), 431–441.

[9] H. Porsche. *Die helios-sonde als experimententräger.* Raumfahrtforschung **19** (1975), 233.

[10] C. Salem, D. Hubert, C. Lacombe, A. Mangenay, D. Larson, and L. R.P. *Electron properties and coulomb collisions in the solor wind at 1 au: Wind observations.* Astrophys. J. **585** (2003), 1147–1157.

[11] R. Schwenn, H. Rosenbauer, and H. Miggenrieder. *Das plasmaexperiment auf helios.* Raumfahrtforschung **19** (1975), 226.

[12] P. Song, X. X. Zhang, and G. Pashmann. *Uncertainties in plasma measurements: effect of lower cutoff energy and spacecraft charge.* Planet. Space Sci. **45** (1997), 255.

Extrakce booleovských pravidel z neuronových sítí pomocí zpětné propagace mnohostěnu

Tomáš Vondra

2. ročník PGS, email: tv@fuzzy.cz
Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská, ČVUT
školitel: Ing. RNDr. Martin Holeňa CSc., Ústav informatiky AV ČR

Abstract. The ability to extract information stored in a trained artificial neural network in the form of logical rules is important for „explanation" of the network's behaviour – the retrieved rules can help us to comprehend how the neural network reaches decisions, determine and understand which properties of the processed data are important, and last but not least the gathered rule sets can be subsequently used in other applications. Usage of artificial neural networks in the field of „mission critical" applications requires a verification of the network, and a rule extraction techniques appear to be a natural choice. However many methods and algorithms from this area to date are based on a heuristics and thus possess some important disadvantages – computational and algorithmic complexity and minimal possibilities to estimate or limit error. The REBAP algorithm, whose analyze and modifications are subject of this paper, is not perfect but seems to be quite promising.

Abstrakt. Schopnost extrahovat ve formě logických pravidel informace uložené v naučené neuronové síti je důležitá pro „vysvětlení" fungování dané neuronové sítě – získaná pravidla nám mohou pomoci pochopit jak se neuronová síť rozhoduje, zjistit a pochopit které vlastnosti zpracovávaných dat jsou důležité, a v neposlední řadě lze takto získaná pravidla dále využívat. Analýza naučené neuronové sítě je důležitá také z hlediska verifikace její správné funkce, což je důležité zejména pro nasazení neuronových sítí v tzv. „mission critical" aplikacích. Mnoho současných metod a algoritmů z této oblasti je však založena na principu heuristiky a jako taková má dva základní nedostatky – vysokou výpočetní náročnost a minimální možnosti odhadu chyby. Jako poměrně slibný se jeví algoritmus REBAP publikovaný, jehož základnímu rozboru a možnostem dalšího vývoje se věnuje tento text.

1 Úvod

Umělé neuronové sítě se během zhruba 60 let, které uplynuly od formulace jejich základů ve 40-tých letech 20. století, staly velice populárním prostředkem umělé inteligence. Z velké míry k tomu přispěla jejich značná flexibilita a relativní jednoduchost použití, stále sofistikovanější metody konstrukce a algoritmy učení, ale také rozmach výpočetní techniky.

Stále širší využití umělých neuronových sítí však přináší i potřebu metod pro jejich efektivní analýzu, zaměřenou zejména na verifikaci jejich funkce neboť neuronové neuronové sítě obecně po naučení fungují v podstatě jako černá skříňka. Často se říká že znalosti a informace získané během fáze učení jsou v neuronové síti uloženy „rozprostřeně" například prostřednictvím vah (synapse) resp. vychýlení (neurony), a jejich další přímé využití je tedy nemožné.

Schopnost získávat ve srozumitelné formě informace nastřádané v neuronové síti během procesu učení je velmi důležitá zejména z následujících dvou (velmi úzce souvisejících důvodů):

- **ověření funkce sítě** – Odpověď na otázku „Nechová se síť pro některé vstupy neočekávaně?" je důležitá zejména pro tzv. mission critical systémy, u kterých by jediná chybná odpověď sítě mohla znamenat velké škody na majetku nebo dokonce na lidských životech.

- **další využití získaných informací** – Přirozenou možností je pochopitelně využití neuronové sítě pouze jako prostředku pro extrakci pravidel z dat, tj. jako nástroje tzv. data-miningu. Například pokud by výstupem analýzy byla logická pravidla je možné jejich využití jako vstupů v různých podpůrných a expertních systémech.

Další důvody k rozvoji metod extrakce pravidel z neuronových sítí a podrobnější vysvětlení těch zde uvedených lze nalézt například v [1].

Následující odstavec 2 obsahuje velmi stručný úvod do umělých neuronových sítí – obsahuje zejména definice základních pojmů používaných dále v textu a uvádí několik ilustračních možností využití neuronových sítí. Odstavec 3 se zabývá obecně extrakcí pravidel z neuronových sítí, přičemž uvádí příklady několika metod, a uvádí několik vlastností které by „ideální" metoda pro extrakci pravidel měla mít. Konečně odstavec 4 se podobněji zabývá algoritmem REBAP popsaným F. Maire v článku [3].

2 Krátký úvod do neuronových sítí

Jak již název napovídá, je předobrazem umělých neuronových sítí nervová tkáň, a z toho plyne také podobná vnitřdní struktura. Umělá neuronová síť je skupina navzájem propojených neuronů, provádějících relativně jednoduché výpočty na základě vstupů z okolních neuronů (viz. obrázek 1).
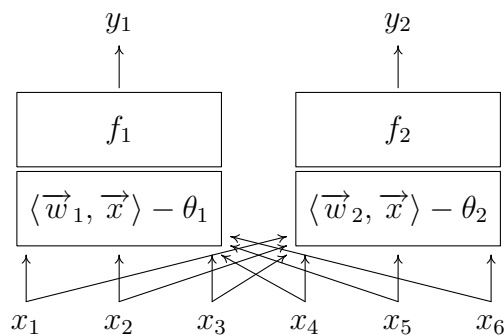
Obrázek 1: Základní elementy umělé neuronové sítě

Jednoduchý výpočet je v každém neuronu realizován aktivační funkcí $f : \mathbb{R} \to \mathbb{R}$ (ne nutně stejnou pro všechny neurony), jejímž vstupem je „celková intenzita" na vstupu neuronu daná $\sum_{i=1}^{n} w_i x_i$ kde $x_1, \ldots, x_n$ jsou neurony ze kterých daný neuron přijímá signály, a váhy $w_1, \ldots, w_n$ reprezentují „útlum" na jednotlivých synapsích. Často je neuronům přiřazován ještě parametr θ interpretovaný jako „vychýlení", a funkce realizovaná v každém neuronu je tedy dána předpisem

$$f \left(\sum_{i=1}^{n} w_i \cdot x_i - \theta \right) = f \left(\langle \overrightarrow{x}, \overrightarrow{w} \rangle - \theta \right) \tag{1}$$

kde $\langle \overrightarrow{x}, \overrightarrow{w} \rangle$ je skalární součin vektorů $\overrightarrow{x} = (x_1, \ldots, x_n)$ a $\overrightarrow{w} = (w_1, \ldots, w_n)$.

V následujícím textu jsou využívány výhradně tzv. vrstevnaté feed-forward neuronové sítě, tj. sítě kde

1. jsou neurony uspořádány do vrstev

2. neurony v první vrstvě jsou tzv. *vstupní neurony*, zastupující „vnější" vstupy sítě

3. neurony v k-té vrstvě mají jako vstupy pouze výstupy neuronů z $k-1$ vrstvy

4. neurony poslední vrstvě jsou tzv. *výstupní neurony*, předávající výstupy mimo neuronovou síť

Vrstevnaté feed-forward sítě tedy neobsahují cykly a každý neuron přijímá pouze informace z vrstvy „těsně pod ním." Uvažujme nyní libovolnou z vrstev neuronové sítě, a zaveďme následující označení:

- m – počet neuronů ve vrstvě pod uvažovanou vrstvou

- n – počet neuronů v uvažované vrstvě

- x_i – výstup i-tého neuronu z vrstvy pod uvažovanou vrstvou ($i \in \widehat{m}$)

- y_i – výstup i-tého neuronu v uvažované vrstvě ($i \in \widehat{n}$)

- f_i – aktivační funkce realizovaná v i-tém neuronu vrstvy ($i \in \widehat{n}$)

- θ_i – vychýlení i-tého neuronu uvažované vrstvě ($i \in \widehat{n}$)

- w_{ij} – váha (zesílení) j-tého vstupu do i-tého neuronu v uvažované vrstvě ($i \in \widehat{n}$, $j \in \widehat{m}$)

Zavedeme-li dále pro označení

$$\overrightarrow{w}_i = \begin{bmatrix} w_{i,1} \\ \vdots \\ w_{i,m} \end{bmatrix} \quad \overrightarrow{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} \quad \overrightarrow{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \quad \overrightarrow{\theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} \tag{2}$$

$$\mathbb{W} = (w_{ij})_{i,j=1}^{n,m} = \begin{bmatrix} \overrightarrow{w_1}^T \\ \vdots \\ \overrightarrow{w_n}^T \end{bmatrix} = \begin{bmatrix} w_{1,1} & \cdots & w_{1,m} \\ \vdots & & \vdots \\ w_{n,1} & \cdots & w_{n,m} \end{bmatrix} \tag{3}$$

realizuje uvažovaná vrstva neuronové sítě zobrazení $\overrightarrow{f} : \mathbb{R}^m \to \mathbb{R}^n$ dané předpisem

$$\overrightarrow{y} = \overrightarrow{f}\left(\mathbb{W}\overrightarrow{x} - \overrightarrow{\theta}\right) = \begin{bmatrix} f_1\left(\langle \overrightarrow{w}_1, \overrightarrow{x}\rangle - \theta_1\right) \\ \vdots \\ f_n\left(\langle \overrightarrow{w}_n, \overrightarrow{x}\rangle - \theta_n\right) \end{bmatrix} \tag{4}$$

Uvažujme nyní neuronovou síť s k vrstvami, m_N vstupními neurony a n_N výstupními neurony. Vstupy sítě (zastupované vstupními neurony) označme $\overrightarrow{x}_N$, výstupy sítě (tj. výstupy poslední vrstvy) označme $\overrightarrow{y}_N$, a zobrazení realizovaná v jednotlivých vrstvách neuronové sítě označme $f^{(1)}, \ldots, f^{(k)}$. Potom celá neuronová síť realizuje zobrazení $\mathbb{R}^{m_N} \to \mathbb{R}^{n_N}$ dané předpisem

$$\overrightarrow{y}_N = \overrightarrow{g}\left(\overrightarrow{x}_N\right) = \overrightarrow{f}^{(k)} \circ \overrightarrow{f}^{(k-1)} \circ \cdots \circ \overrightarrow{f}^{(2)} \circ \overrightarrow{f}^{(1)}\left(\overrightarrow{x}_N\right) \tag{5}$$

Z výše uvedeného je zřejmé že neuronová síť je plně definována zadáním:

- počtu vrstev

- počtu neuronů n v jednotlivých vrstvách

- vah $\mathbb{W}$ a vychýlení $\overrightarrow{\theta}$ pro jednotlivé vrstvy

- aktivačních funkcí f_i realizovaných v neuronech jednotlivých vrstev

Přitom proces učení neuronové sítě spočívá právě v modifikaci vah $\mathbb{W}$ a vychýlení θ tak aby byla minimalizována zvolená chyba – lze tedy říci že právě ve vahách a vychýleních jsou uloženy informace které se neuronová síť „naučila." Metody a algoritmy učení neuronových sítí však nejsou předmětem tohoto článku, předpokládejme tedy že máme k dispozici (dostatečně dobře) naučenou neuronovou síť.

3   Extrakce pravidel z neuronových sítí

V oblasti extrakce pravidel z neuronových sítí dnes existuje mnoho velmi různorodých a obtížně srovnatelných metod – metody generují pravidla různých tvarů, často fungující jen na specifické architektuře neuronové sítě, atd. V článku [1] je uveden návrh na základní taxonomii na základě pěti následujících hledisek:

- výrazová schopnost generovaných pravidel

- využití znalostí o interním uspořádání sítě, tj. architektury a hodnot parametrů

- využití speciálních tréninkových režimů

- kvalita produkovaných pravidel

- algoritmická složitost

Podívejme se nyní na jednotlivé body pro algoritmus REBAP[1], popisovaný blíže v odstavci 4.

K extrakci pravidel lze přistupovat jako k problému inverze, tj. pro síť realizující zobrazení $f : \mathbb{R}^m \to \mathbb{R}^n$ a oblast $Y \subset \mathbb{R}^n$ je úkolem najít takovou oblast $X \subset \mathbb{R}^m$ že

$$x \in X \Leftrightarrow f(x) \in Y \tag{6}$$

V případě extrakce pravidel oblast Y reprezentuje specifické výstupy neuronové sítě – například při využití dané neuronové sítě pro klasifikaci může reprezentovat zařazení do první třídy, tj. tvrzení „aktivní je pouze výstupní neuron č. 1" a podobně. Oblast X potom reprezentuje všechny vstupy klasifikované do první třídy.

Důležitým krokem je zejména volba tvaru oblasti Y, která musí být volena s ohledem na algoritmus samotný a současně musí být dostatečně efektivně (zejména s ohledem na paměťové a výpočetní nároky). V algoritmu REBAP jsou jako oblasti použity mnohostěny, a to zejména z důvodu využití afinních aproximací a jejich jednoduchou reprezentaci.

Jak bude vidět v následujícím odstavci, algoritmus REBAP je použitelný pouze pro vrstevnaté feed-forward neuronové sítě se spojitými a diferencovatelnými aktivačními funkcemi, přičemž je přímo založen na znalosti vnitřních parametrů sítě. Nevyužívá však žádný specifický tréninkový režim, předpokládá pouze že analyzovaná neuronová síť je dostatečně dobře naučena.

Kvalita pravidel je dosti široký pojem, ale kromě jiného ji lze posuzovat z hlediska věrnosti (tj. jak vygenerovaná pravidla odpovídají analyzované síti) a složitosti pravidel. Velkou výhodou algoritmu REBAP je právě schopnost určovat s jakou přesností má být oblast X (resp. její aproximace) počítána. To umožňuje generovat libovolně přesná pravidla, ovšem za cenu zvyšování složitosti pravidel (totiž složitosti oblasti X).

Zvyšování přesnosti však bohužel přináší také problémy u posledního bodu, neboť velmi výrazně zvyšuje algoritmickou (a výpočetní) složitost.

Podrobnější informace o uvažovaných klasifikačních kritériích a zařazení základních metod lze najít v [1].

4 REBAP

Jak již bylo uvedeno v odstavci 3, důležitým krokem při formulaci extrakce pravidel jako problému inverze je volba vhodného tvaru „invertované" oblasti $Y \subset \mathbb{R}^{n_N}$. Tato oblast musí být zvolena tak aby:

1. její výpočetní reprezentace byla co nejefektivnější (s ohledem na paměťové a výkonnostní nároky)

2. při průchodu jednotlivými vrstvami sítě neměnila svou „povahu"

3. umožňovala efektivní aproximaci jiných typů oblastí

---

[1]**R**ule **E**xtraction by **BA**ck-propagation of Polyhedra

Uvažujme nyní obecné reálné afinní zobrazení $T : \mathbb{R}^m \to \mathbb{R}^n$ (7) a mnohostěn $P \subseteq \mathbb{R}^n$ daný systémem $l \in \mathbb{N}$ nerovností (8).

$$T(x) = \mathbb{A}\overrightarrow{x} + \overrightarrow{b} \quad \mathbb{A} \in \mathbb{R}^{n,m}, \ \overrightarrow{b} \in \mathbb{R}^n, \ \overrightarrow{x} \in \mathbb{R}^m \tag{7}$$

$$P = \left\{ \overrightarrow{x} \in \mathbb{R}^n \mid \mathbb{C}\overrightarrow{x} \leq \overrightarrow{d} \right\} \quad \mathbb{C} \in \mathbb{R}^{l,n}, \ \overrightarrow{d} \in \mathbb{R}^l \tag{8}$$

Potom „inverze" mnohostěnu P při zobrazení je dána předpisem

$$T^{-1}(P) = \left\{ \overrightarrow{y} \in \mathbb{R}^m \mid \mathbb{C}\left(\mathbb{A}\overrightarrow{y} + \overrightarrow{b}\right) \leq \overrightarrow{d} \right\} \tag{9}$$

$$= \left\{ \overrightarrow{y} \in \mathbb{R}^m \mid \mathbb{C}\mathbb{A}\overrightarrow{y} \leq \overrightarrow{d} - \mathbb{C}\overrightarrow{b} \right\} \tag{10}$$

Tj. inverze mnohostěnu při afinním zobrazení je opět mnohostěn. Současně platí že mnohostěny jsou velmi oblíbené pro relativní jednoduchost výpočetní reprezentace, a možnost libovolně přesné aproximace oblastí jiného typu (skládání „kostiček"). Mnohostěny jako takové tedy splňují první a třetí kritérium pro volbu tvaru oblastí, ovšem za předpokladu že pracujeme s afinními zobrazeními.

Všimněme si že výpočet (1) realizovaný v jednotlivých neuronech, a tedy i v celé k-té vrstvě (4), má vlastně dvě fáze (viz. obrázek 1):

1. **afinní fázi** danou pro i-tý neuron zvolené vrstvy resp. pro celou zvolenou vrstvu předpisem

$$z_i = \langle \overrightarrow{w}_i, \overrightarrow{x} \rangle - \theta_i \tag{11}$$

$$\overrightarrow{z} = \mathbb{W}\overrightarrow{x} - \overrightarrow{\theta} \tag{12}$$

2. **aktivační fázi** která je obecně nelineární, a je dána předpisem (viz. 4)

$$y_i = f_i\left(\overrightarrow{z}_i\right) \qquad \text{resp.} \qquad \overrightarrow{f}\left(\overrightarrow{z}\right) \tag{13}$$

Aproximací funkce realizované v aktivační fázi funkcí po částech afinní získáme aproximaci neuronové sítě kde každá vrstva provádí afinní transformaci. Nechť f_i je monotonní a diferencovatelná funkce na intervalu $[a, b]$ a $(c_i)_{i=1}^s$ nechť je jeho rozdělení, tj. nechť

$$c_1 = a, \ c_s = b \tag{14}$$

$$c_i < c_{i+1} \quad \forall i \in \{1, \ldots, s-1\} \tag{15}$$

Potom afinní aproximaci φ_i funkce f_i definujme pro $t \in [c_i, c_{i+1}]$ jako

$$\varphi_i(t) = f_i(c_i) + \frac{f_i(c_{i+1}) - f_i(c_i)}{c_{i+1} - c_i}(t - c_i) \tag{16}$$

Tato aproximace je pro funkci $f(x) = 1/(1 + e^{-5(x-1.25)})$ na intervalu $[0.5, 1.75]$ s rozdělením $[0.5, 1, 1.75]$ ilustrována na obrázku 2. Jak je vidět v jednorozměrném případě $f_i : \mathbb{R} \to \mathbb{R}$ se jedná o funkci po částech lineární.

Afinní aproximace $\overrightarrow{\varphi}$ funkce $\overrightarrow{f}$ realizované na zvolené vrstvě sítě (4) je jednoduchým rozšířením vztahu (16). Nechť $\times_{i=1}^m [a_i, b_i] \subset \mathbb{R}^m$ je vícerozměrný interval a $(c_{i,j})_{i,j=1}^{m,s_i}$ je
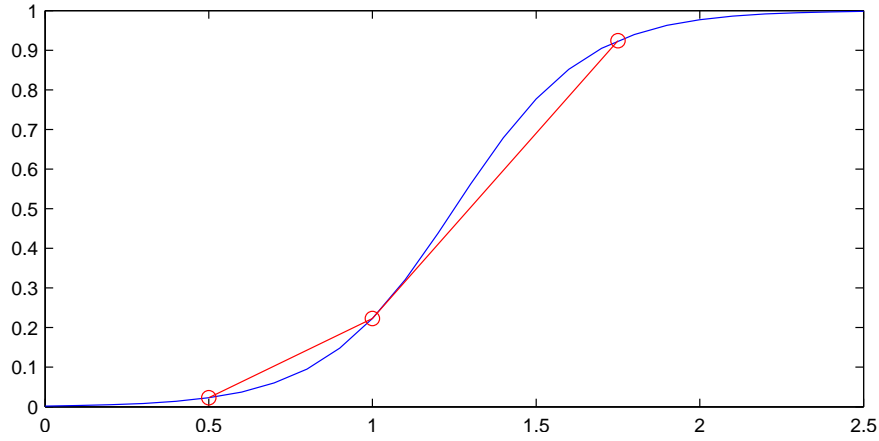
Obrázek 2: Aproximace po částech afinní funkcí

jeho rozdělení, tj. pro každé pevně zvolené $i \in \widehat{m}$, jednorozměrný interval $[a_i, b_i]$ a jeho rozdělení $(c_{i,j})_{j=1}^{s_i}$ platí (14) a (15). Zvolme $\overrightarrow{t} \in \times_{i=1}^{m}[c_{i,j_i}, c_{i,j_i+1}]$ tj. nechť pro složky t_i platí $t_i \in [c_{i,j_i}, c_{i,j_i+1}]$, a zaveďme označení

$$\overrightarrow{c}_j = (c_{1,j_1}, c_{2,j_2}, \ldots, c_{m,j_m}) \tag{17}$$

tj. $\overrightarrow{c}_j$ je vektor tvořený dolními mezemi ovažovaného vícerozměrného intervalu. Potom i−tá souřadnice afinní aproximace $\overrightarrow{\varphi}$ je v bodě $\overrightarrow{t}$ dána předpisem

$$\varphi_i(\overrightarrow{t}) = f_i\left(c_{i,j_i}\right) + \frac{f_i(c_{i,j_i+1}) - f_i(c_{i,j_i})}{c_{i,j_i+1} - c_{i,j_i}}(t_i - c_{i,j_i}) \tag{18}$$

a celou aproximaci $\overrightarrow{\varphi}$ je tedy možno zapsat jako

$$\overrightarrow{\varphi}(\overrightarrow{t}) = \mathbb{D}\,\overrightarrow{t} + \overrightarrow{\delta} \tag{19}$$

kde

$$\mathbb{D} = \begin{bmatrix} \frac{f_1(c_{i,j+1}) - f_1(c_{i,j})}{c_{i,j+1} - c_{i,j}} & & \\ & \ddots & \\ & & \frac{f_{n_k}(c_{i,j+1}) - f_{n_k}(c_{i,j})}{c_{i,j+1} - c_{i,j}} \end{bmatrix} \tag{20}$$

$$\overrightarrow{\delta} = \overrightarrow{f}\left(\overrightarrow{c}_j\right) - \mathbb{D}\,\overrightarrow{c}_j \tag{21}$$

takže budeme-li opět uvažovat mnohostěn P daný systémem (8) je jeho vzor φ^{-1} po průchodu druhou (aktivační) fází výpočtu na zvoleném intervalu dán předpisem

$$\varphi^{-1}(P) = \left\{ \overrightarrow{z} \in \mathbb{R}^n \mid \mathbb{C}\mathbb{D}\overrightarrow{z} \leq \overrightarrow{d} - \mathbb{C}\overrightarrow{\delta} \right\} \tag{22}$$

Provedeme-li dle vztahu (8) ještě zpětnou propagaci tohoto mnohostěnu první (afinní) fází, danou předpisem (12), získáme na každém intervalu rozdělení uvažovaného intervalu polytop daný vztahem

$$\left\{ \overrightarrow{x} \in \mathbb{R}^n \mid \mathbb{C}\mathbb{D}\mathbb{W}\overrightarrow{x} \leq \overrightarrow{d} - \mathbb{C}\overrightarrow{\delta} + \mathbb{C}\mathbb{D}\overrightarrow{\theta} \right\} \tag{23}$$

Příklad 1. Uvažujme jednoduchou neuronovou síť (resp. jednu vrstvu neuronové sítě) s topologií znázorněnou na obrázku 3. Proměnné x_1 a x_2 reprezentují výstupy předchozí
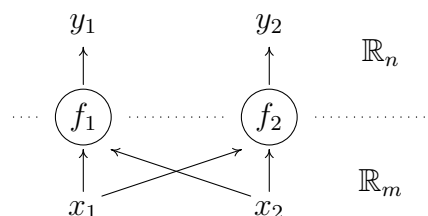
Obrázek 3: jednoduchá neuronová síť

vrstvy, a předpokládejme že v neuronech uvažované vrstvy jsou jako aktivační funkce f_1 a f_2 použity jednoduché sigmoidální funkce, dané obecným předpisem
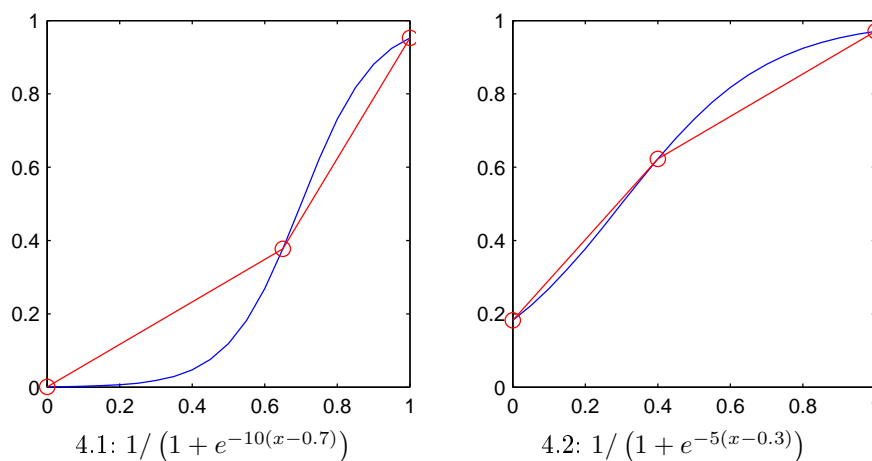
$$\text{sigmf}(x, a, c) = \frac{1}{1 + e^{-a(x-c)}} \qquad x, a, b \in \mathbb{R} \tag{24}$$

a nechť

$$f_1(x) = \text{sigmf}(x, 10, 0.7) \tag{25}$$
$$f_2(x) = \text{sigmf}(x, 5, 0.3) \tag{26}$$

Uvažujme tyto funkce na intervalu $[0, 1] \times [0, 1] \subset \mathbb{R}^2$ a jeho rozdělení na intervaly $[0, 0.65, 1] \times [0, 0.4, 1]$ znázorněné na obrázku 5.2. Tomuto rozdělení odpovídají afinní aproximace znázorněné na obrázku 4, a také rozdělení oboru hodnot, znázorněné na obrázku 5.2.

4.1: $1/\left(1 + e^{-10(x-0.7)}\right)$ 　　　 4.2: $1/\left(1 + e^{-5(x-0.3)}\right)$

Obrázek 4: Aproximace funkcí realizovaných v neuronech
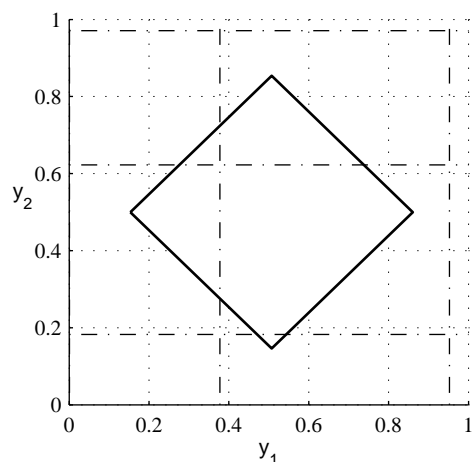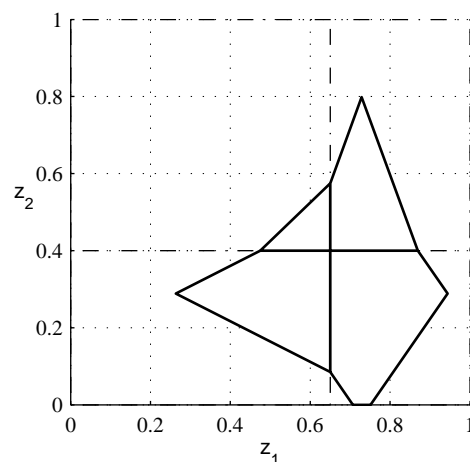
5.1: obor hodnot: P 5.2: definiční obor: $\varphi^{-1}(P)$

Obrázek 5: Zpětná propagace aproximací aktivační fáze a rozdělení definičního oboru a oboru hodnot

Uvažujme nyní mnohostěn P daný systémem nerovnic (27), znázorněný na obrázku 5.1, a proveďme zpětnou propagaci mnohostěnu danou předpisem (23).

$$\mathbb{C} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \\ -1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \leq \begin{bmatrix} 0.3464 \\ 0.3607 \\ -0.6535 \\ 1.3607 \end{bmatrix} \tag{27}$$

Aplikací předpisu (22) postupu na jednotlivé intervaly, získáme sjednocení mnohostěnů $\varphi^{-1}(P)$ znázorněné na obrázku (5.2). Zpětnou propagací jednotlivých složek tohoto sjednocení afinní fází, tj. využitím vztahu (23) získáme $\varphi^{-1}(P)$ tj. zpětnou propagaci mnohostěnu P zvolenou vrstvou neuronové sítě.

Předpokládejme že vychýlení $\overrightarrow{\theta}$ a váhy $\mathbb{W}$ pro zvolenou vrstvu jsou dány takto:

$$\mathbb{W} = \begin{bmatrix} 0.5 & 0.7 \\ 0.9 & -0.1 \end{bmatrix} \qquad \overrightarrow{\theta} = \begin{bmatrix} 0.2 \\ 0.3 \end{bmatrix} \tag{28}$$

Potom vzor mnohostěnu $f^{-1}(P)$, daný na jednotlivých intervalech rozdělení vztahem (22) je znázorněn na obrázku 6.

4.1 Přesnost a složitost algoritmu

Velkou výhodou který algoritmus F. Maire (viz. [3]) má oproti jiným algoritmům uvedeným například v [1], je pojem věrnosti (fidelity), tj. míra chyby mezi množinou pravidel extrahovaných z (aproximované) neuronové sítě a skutečnými výstupy neuronové sítě. Tato míra chyby přímo souvisí s přesností afinní aproximace prováděné při zpětné propagaci mnohostěnu aktivační fází každé vrstvy neuronové sítě.
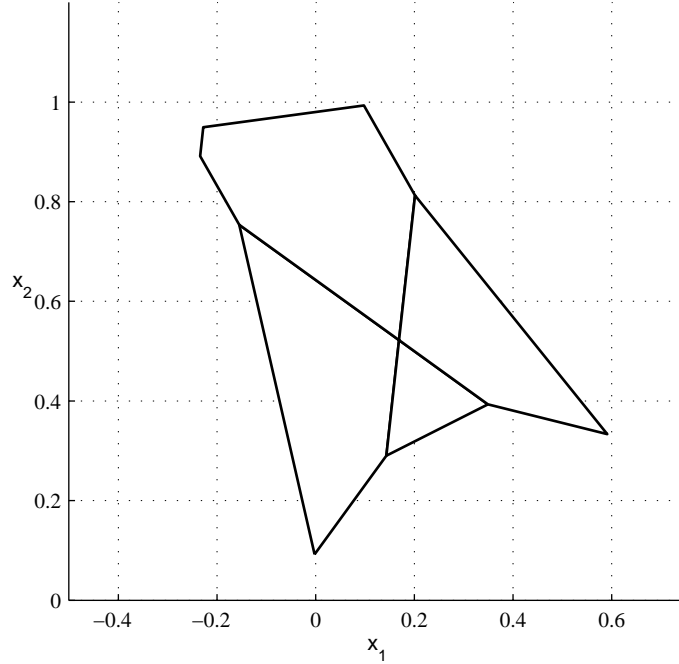
Obrázek 6: získaná aproximace vzoru mnohostěnu $f^{-1}(P)$

Definice 2 (Rozdílnost množin A, B). Nechť $A, B \subseteq \mathbb{R}^n$ jsou libovolné množiny, a $d(x, y)$ je Euklidovská vzdálenost. Potom *míru rozdílnosti* těchto množin $\rho(A, B)$ definujme jako

$$\rho(A, B) = \max \left(\max_{x \in A} d(x, B), \max_{x \in B} d(x, A) \right) \tag{29}$$

kde

$$d(x, A) = \min_{y \in B} d(x, y) \tag{30}$$

Přávě definovaná funkce ρ je vzdálenost (důkaz viz. [3]), a na $\rho(A, B)$ lze pohlížet jako na maximální průměr koule jejíž střed leží v $A \triangle B$ (symetrická diference množin) a která neprotíná $A \cap B$.

Věta 3 (Chyba při aproximaci aktivační fáze). *Nechť f je monotonní diferencovatelná funkce, φ její afinní aproximace a A množina v jejich oboru hodnot. Potom*

$$\rho\left(f^{-1}(A), \varphi^{-1}(A)\right) \;\leq\; \sqrt{\sum_d M_d^2 \alpha_d^4} \tag{31}$$

$$\rho\left(f^{-1}(A), \varphi^{-1}(B)\right) \;\leq\; \frac{1}{m} \rho(A, B) + \rho\left(f^{-1}(B), \varphi^{-1}(B)\right) \tag{32}$$

kde M_d je horní mez druhé derivace funkce f^{-1} na uvažovaném intervalu, a α_d je největší interval rozdělení intervalu dle d-té dimenze – viz. vztah (17) na straně 195, tj.

$$\alpha_d = \max_{i \in \widehat{s_d-1}} |c_{d,i+1} - c_{d,i}| \tag{33}$$

Věta 4 (Chyba při afinní fázi). *Nechť A, B jsou dvě množiny v oboru hodnot afinního zobrazení $T(x) = \mathbb{W}x + \theta$. Potom*

$$\rho\left(T^{-1}(A), T^{-1}(B)\right) \leq \frac{1}{\lambda}\rho(A, B) \tag{34}$$

kde λ je největší singulární hodnota $\mathbb{W}$ (tj. odmocnina vlastního čísla $\mathbb{W}^H\mathbb{W}$).

Dvě právě uvedené věty (konkrétně horní odhady (31) a (34)), jejichž důkazy a další podrobnosti lze najít v [3], postačují pro omezení chyby při zpětné propagaci mnohostěnu jednou vrstvou neuronové sítě, neboť na počátku máme mnohostěn $P \subset \mathrm{R}^n$ pro který chceme provádět zpětnou propagaci. Ten nejdříve „projde" aproximací aktivační fáze dle vztahu (22), a věta 3 říká jak moc se získaná oblast $\varphi^{-1}(P)$ liší od té skutečné $f^{-1}(P)$, přičemž velikost této chyby je prostřednictvím α_d přímo provázána s jemností rozdělení uvažovaného intervalu. Věta 4 potom říká jak se tato chyba zvětšuje při zpětné propagaci afinní fází (která je sama o sobě samozřejmě přesná). Vztah (32) umožňuje omezení chyby při zpětné propagaci více vrstvami sítě.

Bohužel právě spojení přesnosti a věrohodnosti ve vztahu (31) je jedním ze základních úskalí tohoto algoritmu, neboť v zásadě znamená exponenciální nárust počtu mnohostěnů při průchodu každou vrstvou sítě.

Uvažme totiž stejně jako v předchozím textu jednu vrstvu neuronové sítě, a předpokládejme že nastává (pochopitelně velmi nepravděpodobný) případ že rozdělení oboru hodnot, indukované rozdělením intervalu v definičním oboru intervalu je ve všech dimenzích ekvidistantní s krokem $\epsilon = 1/k$ kde $k \in \mathbb{N}$.

Přitom „indukcí rozdělení" intervalu je míněno to že pokud zvolíme rostoucí resp. klesající funkci $f : \mathbb{R} \to \mathbb{R}$ a rozdělení $[c_0, \ldots, c_k]$ intervalu $[a, b]$ v definičním oboru, potom tomuto rozdělení odpovídá „indukované" rozdělení $[f(c_0), \ldots, f(c_k)]$ resp. $[f(c_k), \ldots, f(c_0)]$ jeho obrazu v oboru hodnot.

V tomto případě je tedy obor hodnot rozdělen na stejně velké „kostičky" objemu $1/k^n$. To ale znamená že budeme-li touto vrstvou provádět zpětnou propagaci mnohostěnu P o objemu $Vol(P)$, bude tento zasahovat alespoň $k^n \cdot Vol(P)$ uvažovaných kostiček. Budeme-li tedy v tomto speciálním případě uvažovat zpětnou propagaci n-rozměrné jednotkové krychle, rozpadne se tato při zpětné propagaci aktivační fází na k^n elementů.

Právě uvedený odhad je pochopitelně velmi hrubý, uvedená volba rozdělení je velmi nepravděpodobná, ale důvod exponencielního nárustu počtu mnohostěnů v každé vrstvě je z něj dostatečně patrný. V případě nárůstu počtu mnohostěnů v rámci zpětné propagace celou neuronovou sítí by navíc bylo potřeba uvažovat změnu objemu jednotlivých kostiček, která však úzce souvisí s „lokálním" afinním zobrazením na daném elementu rozdělení.

Možné vylepšení analyzovaného algoritmu představuje následující úvaha – monotonie a spojitost aktivačních funkcí f zajišťuje zachování hranice oblasti, tj. skutečnost že se oblasti nebudou „obracet naruby". V dané situaci tedy stačí provádět zpětnou propagaci pro ty elementy (indukovaného) rozdělení které nejsou přímo obsaženy ve zpracovávaném mnohostěnu. Tato skutečnost je ilustrována na obrázku 7. Uvažujme nyní opět n-rozměrnou jednotkovou krychli, která se při zpětné propagaci rozpadá na k^n elementů (k vrstev v každé dimenzi). V této situaci by stačilo provést zpětnou propagaci pouze pro
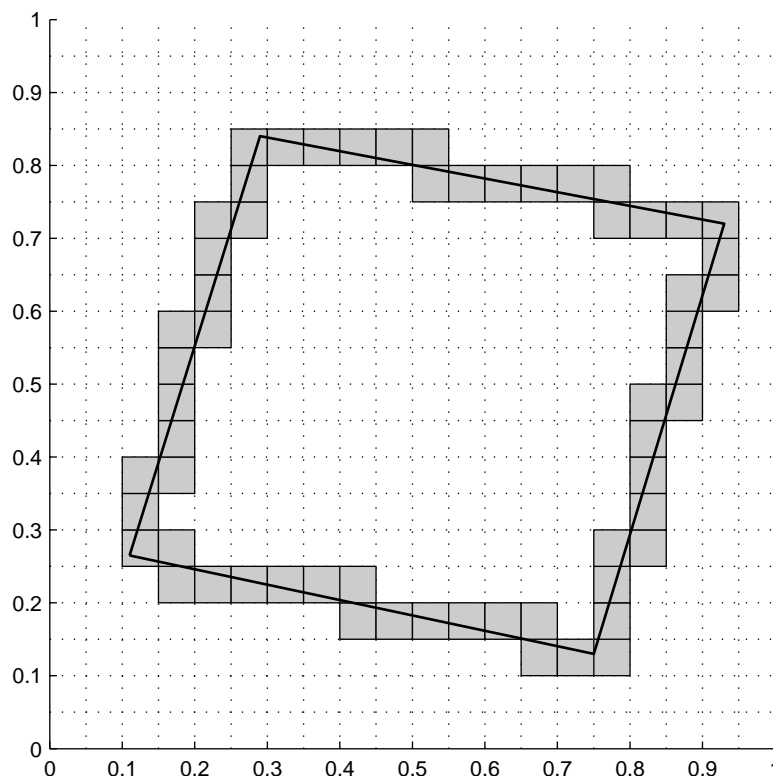
Obrázek 7: Omezení zpětné propagace na „nutné" elementy

„krajní" elementy rozdělení, kterých bude přibližně $2nk^{n-1}$ neboť

$$\underbrace{k^n}_{\text{vše}} - \underbrace{(k-2)^n}_{\text{vnitřní}} = k^n - \sum_{l=0}^{n}\binom{n}{l}(-2)^l k^{n-l} \tag{35}$$

$$= k^n - k^n + 2nk^{n-1} - \underbrace{\sum_{l=2}\binom{n}{l}(-2)^l k^{n-l}}_{>0} \tag{36}$$

$$< 2nk^{n-1} \tag{37}$$

Exponencielní charakter nárustu počtu mnohostěnů tedy zůstává, ale i tak se s ohledem na zpětnou propagaci více vrstami jedná o ne zcela zanedbatelné zlepšení pro praktické aplikace.

Podmínkou efektivity a použitelnosti právě uvedené modifikace je pochopitelně možnost efektivní identifikace „krajních" kostiček – algoritmy pro řešení tohoto problému jsou však známy z výpočetní geometrie.

Další poměrně zajímavou možností modifikace uvažovaného algoritmu je uvolnění podmínky monotonie uvažovaných aktivačních funkcí, tj. zobecnění na libovolné diferencovatelné funkce. Podmínka monotonie totiž znemožňuje použití algoritmu na neuronové sítě s unimodálními aktivačními funkcemi mezi které patří i často používaná Gaussova funkce $\exp(-(x-c)^2/(2\sigma^2))$ nebo zvonovitá funkce $((1+|(x-c)/a|)^{2b})^{-1}$.

5   Závěr

Velkou výhodou algoritmu REBAP oproti jiným metodám, uvedeným například v [1], [2], [5] a [6], je možnost extrakce libovolně přesných pravidel. Tato výhoda s sebou však přináší i nevýhody – kromě složitějších pravidel se jedná zejména o výpočetní náročnost která rychle roste právě v závislosti na požadované rychlosti. Při modifikaci a optimalizaci uvedeného algoritmu zřejmě nelze očekávat zásadní změnu výpočetní náročnosti, avšak i malé změny mohou být v praxi velice užitečné, neboť umožní zpracování větších a/nebo složitějších neuronových sítí.

Lze také zobecňovat uvedený algoritmus, například pro nemonotonní diferencovatelné aktivační funkce, neboť i ty jsou v neuronových sítích velice často používány.

Literatura

[1] Robert Andrews, Joachim Diederich, Alan B. Tickle, „Survey and critique of techniques for extracting rules from trained artificial neural networks," *Knowledge-Based Systems,* Volume 8, Number 6 (December 1995), Elsevier Science B.V., 373–389.

[2] Alan B. Tickle, Robert Andrews, Mostefa Golea, Joachim Diederich, „The truth is in there: directions and challenges in extracting rules from trained artificial neural networks," *IEEE Transactions On Neural Networks,* Volume 9, Number 6 (1998), IEEE, 1057–1068.

[3] Maire F., „Rule-extraction by backpropagation of polyhedra," *Neural Networks,* 12 (1999), Elsevier Science Ltd., 717–725.

[4] Carlos Andújar Gran, „Octree-based Simplification of Polyhedral Solids," *Universidad Politecnica de Catalunya,* Barcelona, 1999.

[5] Ofet Melnik, Jornad Pollack, „Exact Representations from Feed Forward Networks," *Volen Center for Complex Systems, Brandeis University, Waltham, MA, USA,* (April 4, 2000).

[6] Sushmita Mitra, Yoichi Hayashi, „Neuro-Fuzzy Rule Generation: Survey in Soft Computing Framework," *IEEE Transactions On Neural Networks,* Volume 11, Number 3 (May 2000), IEEE, 748–768.