TOWARDS A LINGUISTIC DESCRIPTION OF DEPENDENCIES IN DATA

ILDAR BATYRSHIN*, MICHAEL WAGENKNECHT**

* Institute of Problems of Informatics
Academy of Sciences of Tatarstan and Kazan State Technological University
K. Marx Str. 68, Kazan, 420015, Russia
e-mail: batyr@emntu.kcn.ru

** University of Applied Sciences Zittau/Görlitz
IPM Theodor-Körner-Allee 16, 02763 Zittau, Germany
e-mail: m.wagenknecht@hs-zigr.de

The problem of a linguistic description of dependencies in data by a set of rules R_k: "If X is T_k then Y is S_k" is considered, where T_k's are linguistic terms like *SMALL*, *BETWEEN 5 AND 7* describing some fuzzy intervals A_k. S_k's are linguistic terms like *DECREASING* and *QUICKLY INCREASING* describing the slopes p_k of linear functions $y_k = p_kx + q_k$ approximating data on A_k. The decision of this problem is obtained as a result of a fuzzy partition of the domain X on fuzzy intervals A_k, approximation of given data $\{x_i, y_i\}$, $i = 1, \dots, n$ by linear functions $y_k = p_kx + q_k$ on these intervals and by re-translation of the obtained results into linguistic form. The properties of the genetic algorithm used for construction of the optimal partition and several methods of data re-translation are described. The methods are illustrated by examples, and potential applications of the proposed methods are discussed.

Keywords: fuzzy approximation, linguistic term, fuzzy rule, genetic algorithm

1. Introduction

The problem of a linguistic description of dependencies in data arises in such areas as process monitoring, diagnosis and control, data mining, qualitative reasoning about processes, etc. (Babuska, 1998; Batyrshin *et al.*, 1994; Deogun *et al.*, 1997; Kivikunnas, 1999; De Kleer and Brawn, 1984; Forbus, 1984; Zadeh, 1973; 1975; 1999). Here we consider the problem of the linguistic description of dependencies in given data $\{x_i, y_i\}$, $i = 1, \dots, n$ by a set of rules R_k: "If X is T_k then Y is S_k", where T_k's are linguistic terms like *SMALL*, *LARGE*, *BETWEEN 5 AND 7* describing some fuzzy intervals A_k on X. S_k's are linguistic terms like *DECREASING* and *QUICKLY INCREASING* that characterize the speed of change of y on these intervals.

Fuzzy rules with such right-hand sides (consequent parts) were considered in (Batyrshin, 2002; Batyrshin and Panova, 2001), where the methods of representation of such rules by fuzzy relations were proposed. In the following, we consider the inverse problem: How to generate such a set of fuzzy rules based on given numerical data $\{x_i, y_i\}$, $i = 1, \dots, n$? We propose that a solution to this problem may be obtained as a result of approximation of given data by linear functions $y_k = p_kx + q_k$ on fuzzy intervals A_k (defining an optimal fuzzy partition of the domain of X) and by subsequent re-translation of the result into linguistic form. For the construction of the optimal fuzzy partition a genetic algorithm is applied. Approximating functions are analytically calculated. Retranslation functions describe linguistically the fuzzy intervals A_k and the slopes p_k of approximating linear functions $y_k = p_kx + q_k$ on these intervals. As a result, a rule base is constructed.

The problem considered can be divided into two mutually related parts: (a) construction of a fuzzy partition of the domain of the input variable X on fuzzy intervals with optimal linear approximation of given data on these intervals, and (b) linguistic interpretation of the obtained fuzzy intervals and lines. The first problem is connected with fuzzy piece-wise linear approximation of data, approximation by polygons and splines, shape analysis, fuzzy trend analysis and regression analysis, approximation of data by fuzzy Sugeno models, etc. (Babuska, 1998; De Boor, 1978; Friedman, 1991; Goodrich, 1994; Huarng, 2001; Jang *et al.*, 1997; Kacprzyk and Fedrizzi, 1992; Kivikunnas, 1999; Kosko, 1997; Loncaric, 1998; Wang, 1997; Yu *et al.*, 2001). The second problem is related to the theory of fuzzy information granulation and computing with words (Batyrshin and Panova, 2001; Zadeh, 1997; 1999). The proposed approach to construction of rule bases from given data may be also considered as a calculation of granular derivatives (Batyrshin, 2002).

These problems will be discussed in the following sections. The tackling of the first problem is based on step-wise applications of a genetic algorithm for finding knots, i.e., the points of intersection of fuzzy sets constituting the partition of fuzzy sets, and optimizing the parameters of these sets given a fitness function. Initially, a partition with a sufficiently large number of fuzzy sets which is sequentially reduced till some final optimal partition is chosen. There are three reasons for applying the decremental approach. First, small fuzzy intervals which are not meaningful for linguistic interpretation are deleted. Second, intervals with a small number of given points are deleted since small errors in initial data in such intervals may cause large deviations in slopes. Third, the neighbor intervals with similar linguistic interpretations of dependencies are merged. The application of a basic genetic algorithm (Goldberg, 1989) enables a sufficiently fast solution of the problem considered with easy handling of penalties, criteria and other components.

In the following sections we describe in greater detail the formulation of the problem and methods applied for its solution and illustrate them on examples of linguistic interpretation of given data.

2. Notation and Problem Formulation

Let $D = \{(x_i, y_i)\}$, $x_i \in X$, $y_i \in Y$, $i = 1, \ldots, n$ be a given data set, where X and Y are real intervals, $X = [x_l, x_r]$ for $x_l < x_r$, $Y = [y_l, y_r]$ for $y_l < y_r$. Write $X_D = \{x_i\}$, $i = 1, \ldots, n$ and $Y_D = \{y_i\}$, $i = 1, \ldots, n$. Our goal is to approximate the underlying crisp dependence between input x and output y by a set of rules

$$R_k: \text{ If } X \text{ is } T_k \text{ then } Y \text{ is } S_k,$$

where T_k's are linguistic terms like *SMALL*, *LARGE* and *BETWEEN 5 AND 10* describing some fuzzy intervals (normal and convex fuzzy sets A_k: $X \to [0,1]$). The S_k's are linguistic terms like *DECREASING*, *CONSTANT* and *QUICKLY INCREASING* which describe the speed of the change of the function y on A_k. We will suppose that the linguistic term S_k characterizes the slope p_k of the function $y_k = p_k x + q_k$ that approximates the given data Y_D on A_k.

We will suppose that the membership functions of A_k are continuous, strictly increasing on the left of the kernels $K(A_k)$ and strictly decreasing on the right of the kernels of A_k, which takes place for most popular membership functions (Jang *et al.*, 1997; Klir and Folger 1988). Here, "kernel" means the elements on the real line with membership 1. For brevity, we will denote by $A_k(x)$ the membership value in x. As result, all α-cuts and, particularly, the kernels of fuzzy intervals are closed crisp intervals. Denote by $A_\alpha = [x_{\alpha l}, x_{\alpha r}]$ the α-cut of a fuzzy set A. The value $h_a = x_{\alpha r} - x_{\alpha l}$ is called the *spread* of A_α or the width of A at level α. A fuzzy set is called the *singleton* at $x_0 \in X$ if $A(x) = 1$ at $x = x_0$ and $A(x) = 0$ otherwise.

Definition 1. We say that a set $P = \{A_k\}$, $k = 1, \ldots, m$ of fuzzy intervals on X defines an (α, β)-*fuzzy partition* of X, if

$$\sup_{j \neq k} \Big(\sup_{x \in X} \big((A_j \cap A_k)(x) \big) \Big) = \alpha$$

and

$$\inf_{x \in X} \Big(\Big(\bigcup_{k=1}^{m} A_k \Big)(x) \Big) = \beta,$$

where $0 \le \alpha < 1$, $0 < \beta \le 1$. For the sake of simplicity, (α, β)-fuzzy partitions will be called fuzzy partitions. An (α, β)-partition will be written also as the α-partition.

It follows that the intersection of kernels of fuzzy sets in fuzzy partitions is empty and hence they may be ordered. We will say that a fuzzy interval A_j is located on the left-hand side of a fuzzy interval A_k if $x_u < x_v$ for some $x_u \in K(A_j)$ and $x_v \in K(A_k)$. Such an ordering relation will be denoted by $Aj \prec A_k$. Obviously, this ordering defines a linear strict ordering relation on the classes of fuzzy partitions. We will suppose that in any fuzzy partition $P = \{A_k\}$, $k = 1, \ldots, m$, the indexing of fuzzy intervals corresponds to their ordering, i.e., $A_k \prec A_k + 1$ for all $k = 1, \ldots, m-1$.

We will assume that for all $k = 1, \ldots, m-1$ and for all points x located between the kernel points of A_k and A_{k+1} we have $(A_k \cup A_{k+1})(x) > 0$.

Definition 2. A point x_{0k} is called a *knot* if the membership function of the intersection $A_k \cap A_{k+1}$ of two neighboring intervals of partition P attains its maximal value there. The borders of the set $X = [x_l, x_r]$ will also be denoted as knots: $x_{00} = x_l$ and $x_{0m} = x_r$.

The existence and uniqueness of knot x_{0k} follows from the strict monotonicity of membership functions on both the sides of the kernels of fuzzy intervals.

Definition 3. We call a partition P *normal* if for all $k, j = 1, \ldots, m$ and $k \neq j$ we have $A_k(x) > A_j(x)$ if $x \in (x_{0k-1}, x_{0k})$.

It is clear that any α-partition is normal.

Definition 4. We say that a fuzzy interval A_k contains the points x_i from the initially given data X_D if $x_i \in [x_{0k-1}, x_{0k})$ for $k = 1, \ldots, m-1$ and $x_i \in (x_{0k-1}, x_{0k}]$ for $k = m$. In these cases we will also say that x_i "belongs" to A_k. The set of points from X_D belonging to A_k will be denoted by $X_D(A_k)$. The distance between knots $W(A_k) = x_{0k} - x_{0k-1}$ will be called the *width of the fuzzy set* A_k *in the fuzzy partition* P.

Let $L_X = \{T_1, T_2, \dots\}$ be a set of elements called terms, i.e., L_X is a linguistic variable and its elements will be called the values of L_X (Zadeh, 1975). We will suppose that L_X contains a special element λ denoting the term "*meaningless*."

Definition 5. A function $N_X\colon F(X) \to L_X$, where $F(X)$ is a set of all fuzzy subsets of X, will be called the *re-translation function* and $N_X(A)$ will be called the *name* of a fuzzy set A on X.

For example, the set *TEMPERATURE*$_T = \{$*SMALL, LARGE* $100°$, *BETWEEN* $30°$ *AND* $40°$, *NORMAL, VERY HOT,* $\lambda\}$ defines the values of the linguistic variable *TEMPERATURE*, which may be used for linguistic re-translation of fuzzy sets defined on a set of numerical values of temperature T.

We suppose that the set of values of linguistic variable L_X depends on the possible granulation of the set X which may be determined, e.g., by possible sizes and locations of fuzzy granules on the set X. The elements of L_X may be generated by some grammar and the re-translation function may be defined by some re-translation procedure. We will say that A is a *meaningless* fuzzy set if $N_X(A) = \lambda$; otherwise, it will be called *meaningful*. For example, fuzzy sets may be considered as meaningless if their sizes or locations do not match with possible sizes and locations of fuzzy sets defined by granulation of X in the problem considered.

Definition 6. We call a function $y = px + q$ a *linear approximation of* $D = \{(x_i, y_i)\}$, $x_i \in X$, $y_i \in Y$, $i = 1, \dots, n$ *on a fuzzy set* $A\colon X \to [0, 1]$ if it minimizes the function

$$Q(p, q) = \sum_{i=1}^{n} \big[y_i - y(x_i)\big]^2 A(x_i) \tag{1}$$

for all possible p and q. The parameters p and q can be obtained similarly to the solution of the least-square approximation problem (Conte and de Boor 1972). They can be calculated as follows:

$$p = \frac{KF - JG}{HF - G^2}, \quad q = \frac{HJ - KG}{HF - G^2},$$

where

$$F = \sum_{i=1}^{n} A(x_i), \quad G = \sum_{i=1}^{n} x_i A(x_i),$$

$$H = \sum_{i=1}^{n} x_i^2 A(x_i), \quad J = \sum_{i=1}^{n} y_i A(x_i),$$

$$K = \sum_{i=1}^{n} x_i y_i A(x_i).$$

Let $Z = [p_l, p_r]$ be the set of possible values of slopes p for $y = px + q$ and let $L_Z = \{S_1, \dots, S_t\}$ be a set of terms used for notation of slopes. For example, we may have $L_Z = \{$*QUICKLY DECREASING, DECREASING, SLOWLY DECREASING, CONSTANT, SLOWLY INCREASING, INCREASING, QUICKLY INCREASING*$\}$, or in short form $L_Z = \{$*QDE, DEC, SDE, CON, SIN, INC, QIN*$\}$. The re-translation function $N_Z\colon F(Z) \to L_Z$ defined on the set of all fuzzy subsets of Z will translate numerical values of slopes into linguistic values if these numerical values are considered as singletons.

Definition 7. We call an (α, β)-fuzzy partition *admissible* if it is normal and satisfies the following conditions:

$$K(A_k) \cap X \neq \emptyset, \quad k = 1, \dots, m, \tag{2}$$

$$\alpha \leq \alpha^*, \tag{3}$$

$$\beta \geq \beta^*, \tag{4}$$

$$W(A_k) \geq W, \quad k = 1, \dots, m, \tag{5}$$

$$\mathrm{Card}\,\big(X_D(A_k)\big) > C, \quad k = 1, \dots, m, \tag{6}$$

$$N_Z(p_k) \neq N_Z(p_{k+1}), \quad k = 1, \dots, m-1. \tag{7}$$

Condition (2) requires that at least one kernel point of each fuzzy interval belong to the interval $X = [x_l, x_r]$. Conditions (3) and (4) require the intersection of neighboring fuzzy intervals to be not greater than a given threshold α^*, and the union of fuzzy intervals cover domain X at least at level β^*, where α^*, β^* are fixed numbers from $[0, 1]$ such that $\alpha^* < 1$ and $\beta^* > 0$. Condition (5) is determined by restrictions on a possible granulation of X. If the width of a fuzzy interval is less than the possible width of the minimal granule W, then such a fuzzy interval is considered as meaningless. Condition (6) means that each fuzzy interval contains a sufficient amount of initial data. In the opposite case, small random fluctuations in data may cause large deviations of the slope value p_k in the approximating functions $y_k = p_k x + q_k$. Finally, (7) requires that the linguistic interpretation of slope values on neighboring fuzzy intervals obtained by the re-translation function should be discriminative.

Suppose that re-translation functions N_X and N_Z are defined in a suitable way, an admissible partition $P = \{A_k\}$, $k = 1, \dots, m$ of X is constructed and approximation functions $y_k = p_k x + q_k$ on A_k are determined. Then the re-translation function defines the following rule base:

$$R_k\colon \text{ If } X \text{ is } N_X(A_k) \text{ then } Y \text{ is } N_Z(p_k).$$

That is, the problem of a linguistic description of data D is reduced to the problem of finding an optimal admissible partition of a given set X with respect to some optimality criteria.

We are led to the following problem: Find an admissible fuzzy partition $\{A_k\}$, $k = 1, \dots, m$ of X that minimizes the fitness function

$$Q_t = \sum_{k=1}^{m} \frac{\sum_{i=1}^{n} [y_i - y_k(x_i)]^{2t} A_k(x_i)}{\sum_{i=1}^{n} A_k(x_i)}, \tag{8}$$

where the y_k's are linear approximations of given data $D = \{(x_i, y_i)\}$, $x_i \in X$, $y_i \in Y$, $i = 1, \dots, n$ on fuzzy intervals A_k, m is an unknown number of fuzzy sets and t is some fixed real number used for obtaining better fitting of given data. In our simulations we used $t = 2$.

This problem may be considered as a problem of constraint optimization if conditions (2)–(7) are handled as constraints in the optimization problem. Another approach to this optimization problem is to consider the class of fuzzy partitions satisfying some of the conditions (2)–(7) and to use other conditions as constraints. In the following sections we describe the approach to solve the optimization problem in the class of fuzzy partitions satisfying conditions (2)–(4). We consider 0.5-fuzzy partitions with fuzzy intervals defined by bell membership functions such that the centers of the first and the last membership functions coincide with the left and the right borders of the interval $X = [x_l, x_r]$ and neighboring membership functions intersect at level 0.5. A genetic algorithm is used for obtaining the optimal fuzzy partition minimizing criterion (8). If conditions (5)–(7) are not fulfilled for some fuzzy interval then this interval is merged with one of the neighbor intervals, the number of classes is reduced and the genetic algorithm is applied again. Initially, a partition with a sufficiently large number of fuzzy intervals is chosen. This partition is sequentially reduced till some final optimal partition satisfying (5)–(7) will be obtained.

3. Description of Fuzzy Partitions Used in the Problem

We will consider fuzzy partitions with fuzzy intervals defined as parametric generalized bell membership functions (Jang *et al.*, 1997)

$$A_k(x) = \frac{1}{1 + \left| \dfrac{x - c_k}{a_k} \right|^{2b_k}}, \tag{9}$$

where c_k is the center of a membership function, a_k is its width on the level 0.5 and b_k defines its steepness $(a_k, b_k > 0)$. Moreover, fuzzy partitions $\{A_k\}$, $k = 1, \dots, m$ $(m > 1)$ of $X = [x_l, x_r]$ will be designed such that $c_1 = x_l$, $c_m = x_r$, and $c_{k+1} = c_k + a_k + a_{k+1}$ are fulfilled for all $k = 1, \dots, m-1$, $(m > 2)$, where $a_1 + 2(a_2 + \dots + a_{m-1}) + a_m = (x_r - x_l)$. If $m = 1$, we define $c_1 = 0.5(x_r + x_l)$ and $a_1 = 0.5(x_r - x_l)$. From the construction it follows that all fuzzy intervals intersect at level 0.5 and such partitions will be normal 0.5-partitions satisfying conditions (2)–(4). An example is presented in Fig. 1. As we can see, with increasing b_i the fuzzy sets are changing to "crisp" intervals. This effect can be used to transform many "crisp" constraint optimization problems searching the partition on crisp intervals into a similar fuzzy non-constraint optimization problem based on maximization of parameters b_i. As will be shown later, the solutions may lead to almost crisp fuzzy intervals.

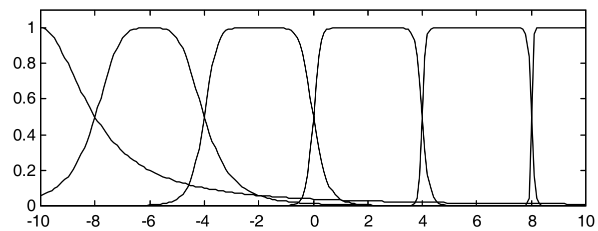

Fig. 1. Fuzzy 0.5-partition of the interval $[-10, 10]$ in 6 fuzzy intervals defined by generalized bell membership functions with parameters $a_i = 2$, $i = 1, \dots, 6$ and parameters b_i having from the left to the right the corresponding values 1, 2, 4, 8, 16, 32. It can be seen that with increasing b the intervals change to "crisp" intervals.

The fuzzy partitions can be determined by parameters a_k and b_k. Optimization of (8) can be performed with respect to these parameters subject to constraints $a_1 + 2(a_2 + \dots + a_{m-1}) + a_m = (x_r - x_l)$ and $a_k, b_k > 0$, $k = 1, \dots, m$. But the optimization problem is simplified if we replace a_k by knots. These can be calculated as follows:

$$x_{0k} = c_k + a_k, \quad k = 1, \dots, m-1, \quad m > 1.$$

The parameters c_k and a_k can be determined as follows:

$$a_1 = x_{01} - x_{00},$$

$$a_k = 0.5(x_{0k} - x_{0k-1}), \quad k = 2, \dots, m-1, \quad m > 2,$$

$$a_m = x_{0m} - x_{0m-1},$$

$$c_1 = x_l,$$

$$c_k = 0.5(x_{0k} + x_{0k-1}), \quad k = 2, \dots, m-1, \quad m > 2,$$

$$c_m = x_r,$$

where $x_{00} = x_l$, $x_{0m} = x_r$, $x_{0k-1} < x_{0k}$, $x_{0k} \in X = [x_l, x_r]$, $k = 1, \dots, m$.

Hence, for the definition of a fuzzy partition of $X = [x_l, x_r]$ into m intervals, it is sufficient to choose $m-1$ knots $x_{01}, \ldots, x_{0m-1}$ within $[x_l, x_r]$ and define m parameters b_k of membership functions.

4. Description of the Genetic Algorithm

We have applied a genetic algorithm for solution of the problem of linguistic approximation of data. As has been mentioned before, for solving this problem special algorithms can be developed like those used in the problem of piece-wise linear or polygon approximations (Friedman, 1991; Goodrich, 1994; Loncaric, 1998; Yu *et al.*, 2001), but the use of genetic algorithms can be motivated by several reasons. First, genetic algorithms constitute a universal tool for solving optimization problems, which can be easily adopted to possible modifications of the optimization problem. Second, we try to give some qualitative description of dependencies in data so we do not need, in general, an exact solution. Moreover, the granulation of linguistic terms used for the re-translation of the obtained fuzzy approximation is usually subjective and tentative. For this reason, in our simulations we do not try to decrease the fitness function (8) as much as possible by increasing the size of the populations generated by the genetic algorithm or by adapting the parameters of the algorithm during the optimization process (Eiben *et al.*, 1999; Goldberg, 1989; Wong and Hamouda, 2000), etc., when we see that the obtained results already facilitate a sufficiently good interpretation. Such an approach is in accordance with the methodology of computing with words tolerant to imprecision for achieving tractability, robustness, a low solution cost and a better rapport with reality (Zadeh, 1996).

As was pointed out in Section 3, a fuzzy partition can be defined by $m-1$ knots $x_{01}, \ldots, x_{0m-1}$ and m parameters $b_1, \ldots, b_m$. The vector $s = (x_{01}, \ldots, x_{0m-1}, b_1, \ldots, b_m)$ with $2m-1$ elements will be called a string. The values $x_{00} = x_l$, $x_{0m} = x_r$ are fixed and are not included in the string. In general, we will assume that $m > 1$. If $m = 1$, then optimization will be performed with respect to one parameter b_1.

Suppose that conditions $x_{0k-1} < x_{0k}$, $x_{0k} \in X = [x_l, x_r]$, and $b_k > 0$, $k = 1, \ldots, m$ are fulfilled. Then the string s defines a fuzzy partition $\{A_k\}$, $k = 1, \ldots, m$ of X into fuzzy intervals and we can calculate for each fuzzy interval the optimal approximations $y_k = p_k x + q_k$ due to (1). As a result, we can calculate the value of the fitness function (8) for the obtained fuzzy partition and approximation functions. Our goal is to find a string s minimizing the fitness function (8). The set of strings considered in some step of the genetic algorithm will be called a population.

Assume that a sufficiently large number of intervals m in fuzzy partitions have been chosen. The genetic algorithm starts with some initial string defining the partition of X and the value of the fitness function for this string is calculated. As such an initial string we may use the parameters of the fuzzy partition of X defined by m equal fuzzy sets (see the top panel in Fig. 2.). Then a population with n_1 strings is randomly generated and the values of the fitness function for each string are calculated. Next, n_2 strings with minimal values of fitness function are selected from the $n_1 + 1$ strings. These "best" strings are called an elite.

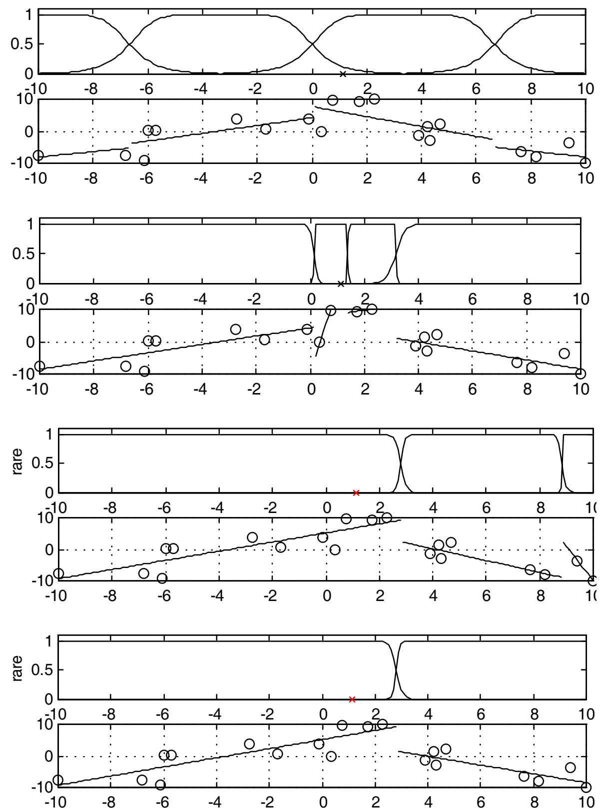

Fig. 2. The set of fuzzy partitions sequentially constructed by the algorithm in Example 1 and the rule base describing the final results: IF X IS LESS THAN 3 THEN Y IS SLOWLY INCREASING; IF X IS GREATER THAN 3 THEN Y IS SLOWLY DECREASING. Parameters of lines: $(p_1, q_1) = (1.4446, 5.1798)$, $(p_2, q_2) = (-1.3869, 5.4134)$.

The strings from the elite are used for generation of new strings. This can be done in various ways. We have used the following method.

For two strings p^k and p^j from the elite (called parents) the set of strings (called children) is generated as follows. First, an index i of knots $(1 \leq i \leq m-1)$ is randomly generated and two new children are obtained as a result of the mutual replacement of knots x_{0i} between parents. Second, a new index i is randomly generated and two new children are obtained as a result of the mutual replacement of all knots with numbers $1, \ldots, i$ between parents. Similar ways of generating of new strings are called a crossover operation (Goldberg, 1989). As a result, the set of $2n_2(n_2 - 1)$ new strings is generated from n_2 elite strings. In both the cases the generation of random integer numbers uniformly distributed on $[1, m-1]$ was used.

Then the following mutation operation was applied to all $n_s = 2n_2(n_2-1)(2m-1)$ elements of new strings. First, n_s normally distributed real numbers were generated. Then all these numbers were normalized to the interval $[-1, 1]$ and multiplied by the parameter M denoting the maximal mutation value. In our simulations we used $M = 0.01(x_r - x_l)$ for all parameter values, but in general this value may be different for knots and parameters b_k. The obtained random mutation values were added to the elements of new strings.

The mutated knot values which did not satisfy the constraints $x_{0k} \in [x_l, x_r]$, were symmetrically mirrored into this interval. Finally, all knots in each new string were sorted to satisfy the requirement $x_{0k-1} < x_{0k}$, $k = 1, \ldots, m-1$.

The obtained set of new strings is called a new population. For each string from the obtained set the value of the fitness function was calculated. The strings of the new population were joined with the elite strings from the previous population and a new elite with n_2 strings having minimal values of the fitness function was chosen.

The genetic algorithm finishes its work after n_3 generations of new populations and selections of new elites. The best string from the final elite with the minimal value of the fitness function is considered as a solution of the optimization problem for given number of fuzzy intervals in fuzzy partitions. The obtained fuzzy partition with m fuzzy intervals satisfies conditions (2)–(4).

Then the fulfilment of conditions (5)–(7) for the fuzzy intervals of the obtained partition is verified. If one of these conditions is violated for some fuzzy interval, then this interval is merged with the neighboring interval and a new partition with $m-1$ intervals is obtained. The string with parameters of the latter is used as the initial string for a new start of the genetic algorithm. The use of the genetic algorithm and the merging procedure are repeated if the obtained fuzzy partition does not satisfy all conditions (5)–(7).

An example of such sequentially obtained fuzzy partitions and corresponding approximations of given data is presented in Fig. 2. The upper partition and corresponding optimal approximations of the criterion (1) are defined by the first initial string with equal fuzzy intervals. The next three partitions were obtained by the genetic algorithm, and merging rare intervals not satisfying condition (6) were used twice. The lower partition and approximations correspond to the final solution of the optimization problem.

5. Merging Procedures

Let us describe the procedure of merging fuzzy intervals of the partition obtained by the genetic algorithm. First, the fuzzy intervals of this partition are tested on the fulfilment of (5). If this condition is not fulfilled for some fuzzy interval, i.e., it is meaningless, then we evaluate the values of the fitness functions of two possible partitions obtained after merging this meaningless interval with left or right neighboring interval. (Of course, if the meaningless fuzzy interval is the first or the last in the partition, then we have no choice.) The parameters of the partition with a smaller value of the fitness function define an initial string for the new start of the genetic algorithm. The new string is obtained from the analyzed string as a result of deleting parameter b_k of the meaningless interval and deleting the knot from the left or the right side of this interval.

If condition (5) is fulfilled for all intervals, then the intervals of the partition are tested to see whether they fulfil condition (6). If this condition is violated for some fuzzy interval, then the choice of the left or the right neighbor, merging intervals and definition of the initial string for starting the genetic procedure are produced similarly to the procedures described above.

If conditions (5) and (6) are fulfilled for all intervals, then the intervals of the fuzzy partition obtained by the genetic algorithm are tested with respect to (7). If this condition is fulfilled for all the fuzzy intervals of the partition, then the optimal partition obtained by the genetic algorithm is considered as a solution of the optimization problem and the partition with approximating lines is re-translated into the set of linguistic rules. This set of rules is considered as the solution of the problem of linguistic description of data dependencies. The re-translation procedures are discussed in the following section.

Suppose that (7) is not fulfilled for some pairs of slopes of approximating lines obtained on neighboring fuzzy intervals of the fuzzy partition, i.e., some pair of neighboring slopes got the same name after application of the re-translation procedure. In this case we analyze all possible new partitions with $m-1$ intervals obtained after merging two neighbor intervals with the same slope name. The partition with the minimal value of the fitness function is chosen for determination of the initial string for the new start of the genetic algorithm. This new string is obtained from the string corresponding to the optimal partition with m intervals as a result of deleting the knot x_{0k}

between merged intervals A_k and A_{k+1} and deleting one of two parameters b_k or b_{k+1} with the smaller value.

6. Re-Translation Procedures

Two re-translation procedures for generation rules from fuzzy partitions and linear approximations are used. These procedures depend on granulation of the set of values of variable X and granulation of the set of possible slope values Z.

Since the rules considered describe the linguistic evaluations of the speed of change of variable Y in some fuzzy intervals of values of variable X, it is supposed that the length of these intervals cannot vanish, and it is bounded from below by some value W (see condition (5)). In the opposite case, such intervals are considered as meaningless. The length of fuzzy intervals may be calculated as the width of the corresponding fuzzy sets at level β if (α, β)-fuzzy partitions are used in the problem of linguistic description of data. It should be noted that the granulation of fuzzy values of X and fuzzy intervals on X may be different. For example, the fuzzy value of X may be represented by a singleton or by a fuzzy set with small width in the rule like "If X is 5 then Y is *SMALL*."

Consider several possible methods of defining the re-translation procedure $N_X\colon\ F(X) \to L_X$. Some of these methods are discussed in (Zadeh, 1975) and in other works on fuzzy sets theory. We have the following:

1. A precisiation function M which makes the meaning of linguistic terms from L_X precise is explicitly given by $M\colon\ L_X \to F(X)$ such that the meaning of the term T is represented by fuzzy set $A = M(T)$. The re-translation procedure N_X can be based on a search for a linguistic term T^* from L_X which has the meaning $M(T^*)$ similar in some sense to the given fuzzy interval A_k, e.g., when some similarity measure $D(A_k, M(T))$ has maximal value for $T = T^*$. We used a similar approach for re-translation of numerical slope functions p_k into linguistic values of the slopes from L_Z.

   Generally, the re-translation procedure may yield several re-translations of a fuzzy set with almost equal similarity values. In this case the possible linguistic re-translations should be given together with the corresponding similarity values. Such a polysemy of re-translation results can be used when the difference between these similarity values is small.

2. A precisiation function M may be given implicitly when the procedure of the calculation of the name of the fuzzy interval A is determined. One of the possible methods of such calculation of names is the following. Some grid $\{G_j\}$ is defined on the interval X and the nodes of this grid are used as the possible ends in the linguistic description of intervals *LESS THAN* G_j, *BETWEEN* G_j *AND* G_k, *GREATER THAN* G_j. In this case the knots of the optimal fuzzy partition may be replaced by the nearest nodes of this grid and these nodes will determine the names of the corresponding fuzzy intervals. It is clear that the name *BETWEEN* G_j *AND* G_k does not determine the crisp interval $[G_j, G_k]$, but it describes some set of fuzzy intervals similar to this crisp interval. In such an interpretation this method of re-translation looks like the previous one.

   The described procedure of re-translation of fuzzy intervals A_k of optimal fuzzy partitions was used in our simulation examples considered below.

3. In the first two methods of re-translation of fuzzy intervals the precisiation function is defined explicitly or implicitly before application of the re-translation procedure. In the third method of the re-translation of fuzzy intervals the precisiation of words used in linguistic description of data is defined during the re-translation procedure. In this case the meaning of words is defined by the fuzzy intervals obtained in the optimal fuzzy partition and denoted by these words. For example, the results of data approximation presented in Fig. 3(c) may be re-translated in the rule base

   IF X IS VERY SMALL THEN Y IS SLOWLY INCREASING,

   IF X IS SMALL THEN Y IS DECREASING,

   IF X IS LARGE THEN Y IS SLOWLY INCREASING,

   IF X IS VERY LARGE THEN Y IS CONSTANT,

   where the meaning of words *VERY SMALL*, *SMALL*, *LARGE* and *VERY LARGE* is determined by fuzzy intervals obtained as a result of the optimization procedure.

7. Examples

We illustrate our approach with two examples. The initial data for these examples were obtained by the program generating piece-wise linear functions with random errors. The generation of such a function is based on the generation of the following values: the number of functions m. The number of points n_k for each function $k = 1, \ldots, m,\ n = n_1 + \cdots + n_m$ values of x, sorted after generation in increasing order, the values of p_k for

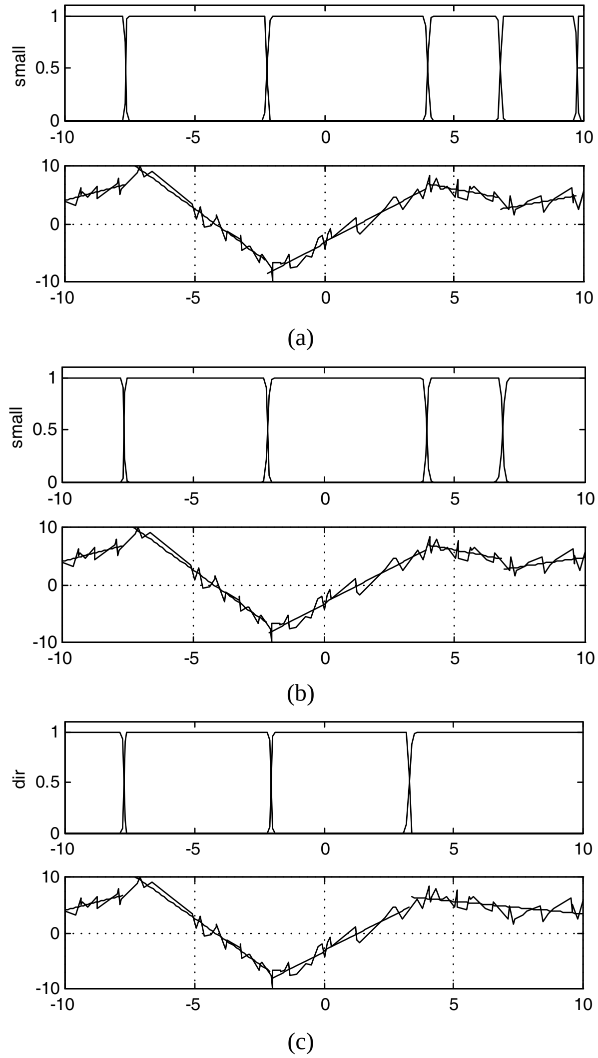

Fig. 3. The last three steps in the linguistic approximation in Example 2: (a) The width of the last fuzzy interval is small and it is merged with the neighboring fuzzy interval, (b) The values of the slopes of the approximating lines on the 4-th and 5-th intervals equal respectively to -0.8571 and 0.6604 and have the same name CONSTANT. Hence, intervals are merged. (c) The values of the slopes at the last step equal to: 1.1939, -3.2649, 2.4231, -0.4542.

each function $y^k = p_k x + q_k$, $k = 1, \dots, m$, the value q_1, and the distance value d between neighboring functions in knots.

After generation of these parameters randomly or deterministically, the sequential calculation of the values of piece-wise linear functions y_i in x_i, $i = 1, \dots, n$, was performed as follows: calculation of n_1 values of y_1 for the first n_1 values of x_i; calculation of parameter $q_2 = y_j - p_2 x_j + d$, where $j = n_1$; calculation of n_2 values of y^2 for the next n_2 values of x_i; calculation of parameter $q_3 = y_j - p_3 x_j + d$, where $j = n_1 + n_2$, etc.

Further, the generation of the value of the maximal deviation and the generation of random values of deviations e_i of functions values y_i are performed. Finally, the values of the functions are calculated as $y_i = y_i + e_i$, $i = 1, \dots, n$.

Appendices 1 and 2 contain two examples of data generated by this program. The results of the linguistic description of these data are presented in Figs. 2 and 3, and they are discussed below. In all simulations we have used the following parameter values:

1. Initial population size: $n_1 = 100$;
2. Number of elite strings: $n_2 = 5$;
3. Number of generations of new populations: $n_3 = 10$;
4. $x_l = -10$, $x_r = 10$, $X = [-10; 10]$;
5. Maximal mutation value: $M = 0.01(x_r - x_l) = 0.2$;
6. Total size of population: $N = 1 + n_1 + 2n_2(n_2 - 1)n_3 = 501$;
7. Minimal width of fuzzy intervals: $W = 1$;
8. Minimal value of points in the interval: $C = 5$;
9. Value of parameter t in (8): $t = 2$;
10. The grid size on X in the second re-translation method: $G = W = 1$, $G_j =$ integer; the values of knots were rounded towards the nearest integer;
11. The linguistic variable $L_Z = \{$*QUICKLY DECREASING, DECREASING, SLOWLY DECREASING, CONSTANT, SLOWLY INCREASING, INCREASING, QUICKLY INCREASING*$\}$ was used for description of the slope values. The granulation of the set of slope values $Z = [-10, 10]$ was defined as the fuzzy 0.5-partition of this interval by 6 generalized bell membership functions with knots $\{-6, -3, -1, 1, 3, 6\}$. The first re-translation method was used. The slope with numerical value p_k obtains the linguistic value S_j of the fuzzy interval to whom p_k belongs with maximal value. When the value of p_k is not far from some knot then it is reasonable to list the names of the neighboring fuzzy intervals in this knot and the membership values of p_k in these intervals.

Example 1 contains a sufficiently small amount of 20 points listed in Appendix 1. We choose $m = 4$ of fuzzy intervals in the initial partition. This example demonstrates the results of the genetic algorithm and the merging procedure which merge the fuzzy interval with a small amount of points ($\mathrm{Card}\,(X_D(A_k)) \le C = 5$) with the neighboring interval (Fig. 2). The linguistic description of data was obtained as a result of application of the first two re-translation procedures described in Section 6, namely

IF X IS LESS THAN 3 THEN Y
IS SLOWLY INCREASING,

IF X IS GREATER THAN 3 THEN Y
IS SLOWLY DECREASING.

Example 2 contains 95 points listed in Appendix 2. Here $m = 8$ was chosen. Figure 3(a) shows the fuzzy partition obtained after triple application of the genetic algorithm with double application of the merging procedure. This partition contains a small fuzzy interval ($W(A_6) < W = 1$). It is merged with the neighboring interval. Next, the partition obtained after application of the genetic algorithm is presented in Fig. 3(b). The last two approximating lines have a "similar direction" because they have been provided with the same name CONSTANT after the re-translation procedure. Figure 3(c) represents the final results obtained after merging the last two intervals and after application of the genetic algorithm. The rules obtained as a result of application of the first two re-translation procedures described in Section 6 are of the following form:

IF X IS LESS THAN -8 THEN Y
IS SLOWLY INCREASING,

IF X IS BETWEEN -8 AND -2
THEN Y IS DECREASING,

IF X IS BETWEEN -2 AND 3
THEN Y IS SLOWLY INCREASING,

IF X IS GREATER THAN 3 THEN Y
IS CONSTANT.

Another rule base representation of results obtained by the third re-translation procedure was discussed in Section 6.

8. Conclusions

A novel approach to linguistic description of dependencies in data is discussed. It is based on the following steps:

- construction of an optimal partition of the domain of the input variable on fuzzy intervals,
- linear approximation of data on these intervals, and
- linguistic interpretation of the obtained intervals and slopes of the approximating lines.

The resulting linguistic description is presented by a rule set like "If X is BETWEEN 5 AND 10 then Y is QUICKLY INCREASING" and "If X is SMALL then Y is DECREASING". The rule base is obtained as a solution of an optimization problem. To solve this problem a genetic algorithm to get optimal partitions and a procedure for merging improper fuzzy intervals were developed. Several methods of re-translating the resulting piece-wise linear approximation into a rule base are considered. Some examples are given for illustration.

To solve the optimization problem, we applied a genetic algorithm due to the following advantage: it is a universal method to search for global solutions in various optimization problems, it is sufficiently easy to implement, it can be easily adopted to changing criteria, penalties and constraints, it enables at least good solutions.

Our approach can be used for the calculation of granular derivatives of functional and statistical dependencies. It can be applied to the linguistic description of trends in time series, systems monitoring, data mining, etc. It enables the construction of knowledge bases with qualitative description of dependencies between variables of technological processes, and it can be applied to qualitative reasoning about systems.

Acknowledgements

This work was partly supported by a research grant of the Saxon Ministry for Sciences and Culture, Germany.

References

Babuska R. (1998): *Fuzzy Modeling for Control*. — Boston: Kluwer.

Batyrshin I. (2002): *On granular derivatives and solution of granular initial value problem*. — Int. J. Appl. Math. Comp. Sci. (in this issue).

Batyrshin I. and Panova A. (2001): *On granular description of dependencies*. — Proc. 9th Zittau Fuzzy Colloquium, Zittau, Germany, pp. 1–8.

Batyrshin I., Zakuanov R. and Bikushev G. (1994): *Expert system based on algebra of uncertainties with memory in process optimization*. — Proc. 1st Int. FLINS Workshop *Fuzzy Logic and Intelligent Technologies in Nuclear Science*, Mol, Belgium, 1994, New York, World Scientific, pp. 156–159.

Conte S.D. and de Boor C. (1972): *Elementary Numerical Analysis. An Algorithmic Approach*. — Tokyo: McGraw-Hill.

De Boor C. (1978): *A Practical Guide to Splines*. — New York: Springer-Verlag.

De Kleer J. and Brawn J. (1984): *A qualitative physics based on confluences*. — Artif. Intell., Vol. 24, pp. 7–83.

Deogun J.S., Raghavan V. V., Sarkar A. and Sever H. (1997): *Data mining: Research trends, challenges, and applications*, In: Roughs Sets and Data Mining: Analysis of Imprecise Data (T.Y. Lin and N. Cercone, Eds.). — Boston, MA: Kluwer, pp. 9–45.

Eiben A.E., Hinterding R. and Michalewicz Z. (1999): *Parameter control in evolutionary algorithms*. — IEEE Trans. Evol. Comput., Vol. 3, No. 2, pp. 124–141.

Forbus K.D. (1984): *Qualitative process theory*. — Artif. Intell., Vol. 24, pp. 85–168.

Friedman J.H. (1991): *Multivariate adaptive regression splines*. — Ann. Stat., Vol. 19, p. 1.

Kacprzyk J. and Fedrizzi M., Eds. (1992): *Fuzzy Regression Analysis*. — Warsaw: Omnitech Press.

Goldberg D.E. (1989): *Genetic Algorithms in Search, Optimization and Machine Learning*. — Ontario: Addison-Wesley.

Goodrich M.T. (1994): *Efficient piecewise-linear function approximation using the uniform metric*. — Proc. 10-th ACM Symp. *Computational Geometry (SCG)*, pp. 322–331.

Huarng K. (2001): *Effective lengths of intervals to improve forecasting in fuzzy time series*. — Fuzzy Sets Syst., Vol. 123, pp. 387–394.

Jang J.S.R., Sun C.T. and Mizutani E. (1997): *Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence*. — New York: Prentice Hall.

Kivikunnas S. (1999): *Overview of process trend analysis methods and applications*. — Proc. Workshop *Applications in Chemical and Biochemical Industry*, Aachen, Germany.

Klir G.J. and Folger T.A. (1988): *Fuzzy Sets, Uncertainty, and Information*. — London: Prentice-Hall.

Kosko B. (1997): *Fuzzy Engineering*. — Upper Saddle River, NJ: Prentice-Hall.

Loncaric S. (1998): *A survey of shape analysis techniques*. — Pattern Recogn., Vol. 31, No. 8, pp. 983–1001.

Wang Li-X. (1997): *A Course in Fuzzy Systems and Control*. — Upper Saddle River, NJ: Prentice Hall.

Wong S.V. and Hamouda A.M.S. (2000): *Optimization of fuzzy rules design using genetic algorithm*. — Adv. Eng. Softw., Vol. 31, pp. 251–262

Yu J.-R., Tzeng G.-H. and Li H.-L. (2001): *General fuzzy piecewise regression analysis with automatic change-point detection*. — Fuzzy Sets Syst., Vol. 119, pp. 247–257.

Zadeh L.A. (1973): *Outline of a new approach to the analysis of complex systems and decision processes*. — IEEE Trans. Syst. Man. Cybern., Vol. 1, pp. 28–44.

Zadeh L.A. (1975): *The concept of a linguistic variable and its application to approximate reasoning*. — Part I: Inf. Sci., Vol. 8, pp. 199-249; Part II: Inf. Sci., Vol. 8, pp. 301–357; Part III: Inf. Sci., Vol. 9, pp. 43–80.

Zadeh L.A. (1996): *Fuzzy logic = computing with words*. — IEEE Trans. Fuzzy Syst., Vol. 4, No. 2, pp. 103–111.

Zadeh L.A. (1997): *Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic*. — Fuzzy Sets Syst., Vol. 90, pp. 111–127.

Zadeh L.A. (1999): *From computing with numbers to computing with words—from manipulation of measurements to manipulation of perceptions*. — IEEE Trans. Circ. Syst. – 1: Fund. Th. Applic., Vol. 45, No. 1, pp. 105–119.

Appendices

A. Data for Example 1 (n=20)

x	y	x	y
-10.0000	-7.7509	1.7414	9.3872
-6.8267	-7.4609	2.2874	10.0000
-6.1241	-9.0711	3.8801	-1.5057
-6.0020	0.1509	4.2416	1.4006
-5.7255	0.2243	4.2981	-2.8755
-2.7265	3.6550	4.7007	2.1011
-1.6841	0.6899	7.6622	-6.6539
-0.1373	3.6132	8.1907	-8.2029
0.3609	-0.2915	9.4199	-3.6266
0.7517	9.7924	10.0000	-10.0000

B. Data for Example 2 (n=95)

x	y	x	y	x	y	x	y
−9.4471	−38.5425	−3.4288	−55.4505	0.2733	−59.1599	5.2626	−37.0734
−9.0527	−40.6865	−3.3944	−51.2637	0.3600	−52.7187	5.5896	−38.2003
−8.8474	−33.1958	−2.8965	−54.9816	0.4620	−50.5749	5.7167	−32.3541
−8.8222	−36.0485	−2.8809	−53.5095	0.5419	−54.3786	5.7358	−33.5486
−8.7887	−34.9089	−2.8430	−59.4271	1.4138	−45.9641	5.8350	−32.1922
−8.5806	−37.2669	−2.5869	−57.6790	1.4996	−51.5577	6.3068	−37.4321
−8.2679	−32.4419	−2.5319	−57.6491	1.5529	−52.1622	6.5710	−38.7950
−8.2600	−37.5136	−2.1643	−64.9882	1.6095	−52.5930	6.6178	−34.7479
−7.4741	−31.0294	−2.1484	−61.9582	2.5358	−41.5024	7.1640	−42.3533
−7.4346	−28.8831	−2.0730	−61.2240	2.8413	−37.2804	7.2260	−36.3245
−7.4044	−35.6157	−1.7430	−66.7916	2.9571	−37.2938	7.3733	−44.3554
−7.3604	−33.6554	−1.6694	−72.8488	3.1874	−42.1781	7.4070	−42.0175
−6.7402	−26.6087	−1.6540	−65.0170	3.7524	−33.6750	7.6902	−40.3082
−6.6560	−23.7808	−1.3881	−64.9923	3.9391	−37.7838	7.8643	−38.8957
−6.4721	−28.5493	−1.3631	−65.5904	4.1727	−27.6648	8.3608	−36.2227
−6.2056	−25.8216	−1.2069	−64.9889	4.2134	−31.4800	8.4450	−43.2889
−5.4412	−33.0609	−1.0831	−61.5660	4.2591	−35.0109	8.6460	−40.0743
−4.7016	−39.7065	−1.0197	−67.0000	4.2843	−32.5297	8.8499	−37.5457
−4.5528	−46.2467	−0.7675	−66.4540	4.4596	−29.0383	9.5023	−33.2119
−4.3631	−41.1609	−0.3966	−61.9568	4.5934	−33.5788	9.5600	−39.1880
−4.2147	−49.4907	−0.2415	−62.3230	4.6729	−33.7012	9.5699	−35.8909
−3.9760	−49.1813	−0.1815	−60.5924	4.8522	−32.2902	9.7543	−42.2540
−3.7973	−45.6884	0.0545	−54.2170	5.1393	−37.0249	9.9597	−34.1132
−3.7969	−44.6787	0.1574	−52.9893	5.2392	−29.6977		