Algebraic Methods for Designing Algorithms for Pattern Recognition and Forecasting[1]

Yu. I. Zhuravlev

Scientific Council on Cybernetics, Russian Academy of Sciences, and Computer Center, Russian Academy of Sciences, ul. Vavilova 40, Moscow, 117967 Russia
e-mail: zhur@ccas.ru

Abstract—An algebraic approach to designing correct pattern recognition algorithms is expanded for a wider class of finite problems. The examples of algebraic constructions for the forecasting and integer linear programming are presented.

Any algorithm that solves a problem can be thought of as a mapping of input data (initial information) onto an answer, i.e., the value of the solution.

$$\text{Initial information} \xrightarrow[\text{mapping}]{\text{algorithm}} \text{answer}$$

As is known, we can define algebraic operations over mappings. In general, these operations yield new mappings (algorithms). These new mappings can have new useful properties. For example, a given set of algorithms may not contain an algorithm that gives an exact solution to a problem, whereas the closure by using certain operations can give a new mapping (algorithm) that ensures the exact solution to the problem.

To make the proposed approach efficient, we need a formal description of the set of allowable input data and responses.

This approach was implemented for the algorithms that are used in pattern recognition on the base of templates [1–3]. It provides an opportunity to develop exact and efficient algebraic recognition methods.

It was found that similar constructions are possible for a much wider case of so-called *finite* problems.

To define natural operations over mappings-algorithms, it is sufficient to define a strong constraint over the set of answers. No constraints on the set of initial information are imposed.

Suppose that the answer in the considered set of problems $\{z\}$ is a vector of a fixed length l. The components 1, 2, …, l of the vector are the answers to the questions $Q_1, Q_2, \ldots, Q_l$; the questions are fixed for all the problems z from $\{z\}$. Hence, the solution to each problem z can be represented as a vector $d(z) = (d_1(z), d_2(z), \ldots, d_l(z))$, where $d_i(z)$ is the answer to the question Q_i in the problem z, $i = 1, 2, \ldots, l$.

[1] This work was supported by the Russian Foundation for Basic Research, project no. 98-07-90180.

Received July 24, 1999

Definition 1. Problems z from $\{z\}$ are called finite if the set of admissible values $d_i(z)$ is finite or denumerable for each $i = 1, 2, \ldots, l$.

Here are examples of finite problems:

Pattern recognition problem. A set M of acceptable objects is a sum of a finite number of possibly overlapping classes $K_1, K_2, \ldots, K_l$. Some information $I(M, K_1, \ldots, K_l)$ about $M, K_1, \ldots, K_l$ along with description of some S within M are given. A pair $(S, I(M, K_1, \ldots, K_l))$ forms a problem z. Its solution is $Q(z) = Q(S) = (d_1(z), d_2(z), \ldots, d_l(z))$, where $d_j(z)$ is the response to the question "does S belong to K_j?", $j = 1, \ldots, l$. In the classical case, the answer has three possible values: 1 (yes), 0 (no), and Δ (don't know). There are other problems with a wider range of possible responses: 0 (no), 1 (unlikely), 2 (likely), 3 (very likely), 4 (yes), and Δ (don't know). More examples can be easily constructed.

Forecasting problem. Information about a process in some phase space is given for some time interval $[0, t + \Delta t]$. A set M of possible states of the process at the moment $t + \Delta t$ is described. A realization R of the process on the interval $[0, t]$ is also given. The set M is decomposed into subsets $K_1, \ldots, K_l$. For example, if $m = [0, 1)$ then the subsets can be $K_1 = [0, \varepsilon)$, $K_2 = [\varepsilon, 2\varepsilon)$, …, $K_l = [(l-1)\varepsilon, l\varepsilon = 1)$, $\varepsilon = 1/l$, $\varepsilon > 0$. The set O_j consists of l questions {"is the state of realization R at the moment $t + \Delta t$ within the class K_j?"}, $j = 1, \ldots, l$. The simplest version supposes the answers "yes," "no," and "don't know."

Integer linear programming problem. Under the constraints

$$\sum_{j=1}^{n} a_{ij}x_j \le b_i, \quad i = 1, \ldots, m,$$

find

$$\max \sum_{j=1}^{n} c_j x_j,$$

where x_j are integers, $j = 1, \ldots, n$.

The questions $Q_1, \ldots, Q_n$, $Q_j(x_j)$ are "what is the optimal value of x_j?" with $j = 1, \ldots, n$. Acceptable answers are $0, -1, +1, \ldots, -k, +k, \ldots, \Delta$ (don't know).

To simplify the task, we assume that the problem has a unique optimal plan.

Obviously, any algorithm A for solving a finite problem z is a mapping of input information $I(z)$ onto the vector $D(z) = (d_1(z), \ldots, d_l(z))$ of answers to the fixed questions. The set of acceptable $d_j(z)$ is at most denumerable, $j = 1, \ldots, l$. The following principal statement holds.

Theorem. Each algorithm-mapping $A(z)$ for a finite problem z can be represented as a composition of mappings B and C

$$A(z) = B \circ C,$$
$$B(z) = (a_1(z), \ldots, a_l(z)) = \bar{a}(z),$$
$$C(\bar{a}(z)) = c_1(a_1(z)), \ldots, c_l(a_l(z)),$$

where $a_i(z)$ are numbers and $i = 1, \ldots, l$.

Similar to the algebraic theory of recognition algorithms, we call mappings B *operators*, and mappings C, *decision rules*.

The rules C_i in the main theorem are the simplest decision rules. Thus, if the domain of possible values of the answers d_i is $d_{i1}, \ldots, d_{ik}$ then c_i can be defined as a rule

$$c_i(a) = d_{it} \text{ if } i \le a \le i+1, \quad i = 1, \ldots, k.$$

The set of algorithms $\{A\}$ generates a set of operators $\{B\}$ and a set of decision rules $\{C\}$. Similar to [1–3], operations over operators are defined. If B_1, B_2, and B are operators, $B_q(z) = \bar{a}_q(z) = (a_1^q, \ldots, a_l^q)$, $q = 1, 2$, $B(z) = a(z) = (a_1, \ldots, a_l)$, and d is constant, then

$$(B_1 + B_2)(z) = \tilde{B}(z) = \bar{a}^1(z) + \bar{a}^2(z) \\ = (a_1^1(z) + a_1^2(z), \ldots, a_l^1(z) + a_l^2(z)), \quad (1)$$

$$(dB)(z) = \hat{B}(z) = d\bar{a}(z) = (da_1(z), \ldots, a_l(z)), \quad (2)$$

$$(B_1B_2)(z) = \breve{B}(z) = \bar{a}^1(z) \circ \bar{a}^2(z) \\ = (a_1^1(z)a_1^2(z), \ldots, a_l^1(z)a_l^2(z)). \quad (3)$$

The operators $\tilde{B}$, $\breve{B}$, and $\hat{B}$ are called *the sum, the product of the operators B_1 and B_2*, and *the product of B and the scalar d*, respectively. Operations (1)–(3) are associative and commutative; operation (1) is distributive with respect to (2) and (3).

The closures $L\{B\}$ and $\mathfrak{A}\{B\}$ of the set of operators $\{B\}$ with respect of operations (1), (2), and (1)–(3) are called *linear* and *algebraic* closures of $\{B\}$, respectively.

The set $L\{B\}$ consists of all possible linear forms over the operators from $\{B\}$, and the set $\mathfrak{A}\{B\}$ consists of all possible polynomials over the operators from $\{B\}$.

The sets $\mathfrak{A}\{A\} = \{B\}\{C\}$, $L\{A\} = L\{B\}\{C\}$ are called the *algebraic* and *linear* closure of the set of algorithms $\{A\}$.

Similar to the recognition problems [1–3], efficient and optimal algorithms (in terms of exactness) for solving finite problems can be constructed in $\mathfrak{A}\{A\}$ and sometimes in $L\{A\}$.

The construction has the following stages.

(1) A finite number of algorithms $A_1, \ldots, A_r$ (usually heuristic) for solving the given class of finite problems is selected.

(2) For each A_i, $i = 1, \ldots, r$, all (or some) constants are replaced by parameters. The domain of parameters is selected to ensure A_i remains sensible. Then, each A_i generates a set (a parametric model) of algorithms $\mathfrak{M}(A_i) = \mathfrak{M}_i$, $i = 1, \ldots, r$.

(3) A set (model) $\mathfrak{M} = \bigcup_{i=1}^{r} \mathfrak{M}_i$ is constructed. The algorithms A from $\mathfrak{M}$ are represented in the form $A = BC$. A set of operators $\mathfrak{M}^1$ and a set of decision rules $\mathfrak{M}^2$ are formed.

(4) The closures $\mathfrak{A}\{\mathfrak{M}^1\}$ and $L\{\mathfrak{M}^1\}$, $\{\mathfrak{A}\} = \{\mathfrak{M}^1\}\{\mathfrak{M}^2\}$, $L\{\mathfrak{M}\} = L\{\mathfrak{M}^1\}\{\mathfrak{M}^2\}$, are constructed.

(5) In the closures $\mathfrak{A}\{\mathfrak{M}\}$ and $L\{\mathfrak{M}\}$, algorithms that are optimal in terms of exactness are constructed for solving finite problems, including different types of recognition and forecasting problems.

REFERENCES

1. Zhuravlev, Yu.I., Correct Algebras over Sets of Incorrect (Heuristic) Algorithms, I, II, III, *Cybernetics*, 1977, no. 4, pp.14–21; 1977, no. 6, pp. 21–27; 1978, no. 2, pp. 35–43.
2. Zhuravlev, Yu.I., On the Algebraic Approach to the Solution of Recognition and Classification Problems, in *Problemy kibernetiki* (The Problems of Cybernetics), Moscow: Nauka, 1978, no. 33, pp. 5–68.
3. Zhuravlev, Yu.I., An Algebraic Approach to Recognition or Classification Problems, *Pattern Recognit. and Image Anal.*, 1998, vol. 8, no. 1, pp. 59–100.

Yuri I. Zhuravlev. Born 1935. Graduated from the Faculty of Mechanics and Mathematics of Moscow State University in 1957. Received a PhD (Kandidat Nauk degree) in 1959 and a Doctoral degree (Doktor Nauk degree) in physics and mathematics in 1965. Since 1969, with the Computer Center of the Russian Academy of Sciences, first, as a Laboratory Head and then, as a Deputy Director. Professor of Moscow State University. A full member of the Russian Academy of Sciences (since 1992) and of the Academy of Sciences of Spain (since 1993). Scientific interests: mathematical logic; algebra; discrete optimization; pattern recognition; and the use of mathematical and computational methods for solving applied problems of data processing and research automation in industry, medicine, geology, and sociology. Author of more than 170 publications on information technology and applied mathematics. Editor-in-Chief of the journal Pattern Recognition and Image Analysis and the yearbook "Raspoznavanie, klassifikatsiya, prognoz: matematicheskie metody i ikh primenenie" (Recognition, Classification, and Forecast: Mathematical Methods and Their Application) and a member of the editorial boards of several international and Russian journals on information technology and applied mathematics.