References

1. Clifton, C., G. Leavens, C. Chambers, T. Millstein, 2000. MultiJava: Modular Open Classes and Symmetric Multiple Dispatch for Java. OOPSLA'00, Minneapolis, Minnesota, October 2000, ACM Press, New York, 130-145. http://www.cs.iastate.edu/~cclifton/multijava/papers/TR00-06.pdf
2. Findler, R., M. Flatt, 1999. Modular Object-Oriented Programming with Units and Mixins. ACM SIGPLAN International Conference on Functional Programming (ICFP '98), 34(1), 94-104. http://www.cs.utah.edu/plt/publications/icfp98-ff/icfp98-ff.pdf
3. Flatt, M., 1999. Programming Languages for Reusable Software Components. PhD thesis, Rice University, Houston, Texas.
http://cs-tr.cs.rice.edu/Dienst/UI/2.0/Describe/ncstrl.rice_cs/TR99-345/
4. Krishnamurthi, S., M. Felleisen, D. P. Friedman, 1998. Synthesizing Object-Oriented and Functional Design to Promote Reuse. ECOOP'98, Brussels, Belgium, July 1998, Springer, Berlin, 91-113.
5. Kühne, T., 1997. The Translator Pattern - External Functionality with Homomorphic Mappings. In Ege, R., M. Singh, and B. Meyer (Eds.), The 23rd TOOLS conference USA 1997, 48-62.
6. Palsberg, J., C. B. Jay, 1997. The Essence of the Visitor Pattern. Technical Report 05, University of Technology, Sydney, Australia.
7. Radenski, A., 2002. Anomaly-Free Component Adaptation with Class Overriding, Journal of Systems and Software, Elsevier Science (under print).
8. Zenger, M., M. Odersky, 2001. Extensible Algebraic Datatypes with Defaults. International Conference on Functional Programming, ICFP 2001, Firenze, Italy, September, 2001. http://lamp.epfl.ch/~zenger/papers/icfp01.pdf
9. Watt, D., D. Brown, 2000. Programming Language Processors in Java: Compilers and Interpreters, Prentice Hall, New York, New York.

Author Information

Atanas Radenski - Chapman University, One University Drive, Orange, California 92866, USA
radenski@computer.org; http://www.chapman.edu/~radenski/

FRONTAL SOLUTIONS: AN INFORMATION TECHNOLOGY TRANSFER TO ABSTRACT MATHEMATICS

V. Jotsov

Abstract: The paper introduces a method for dependencies discovery during human-machine interaction. It is based on an analysis of numerical data sets in knowledge-poor environments. The driven procedures are independent and they interact on a competitive principle. The research focuses on seven of them. The application is in Number Theory.

Keywords: knowledge discovery and data mining, modeling, Number Theory.

1. Introduction

The offered research has begun since 1986 after the exploration of some of the early D. Lenat's papers [Lenat 1976, Lenat 1983]. They gave us the conviction, that the information technologies (IT) are suitable for applications in models which are bounded by Number Theory. The newest evolutionary programming (EP) [EAEA 1997, EA 1997, Nordin 1999] research confirms the possibilities for elaborating new formulas. The considered paper follows the line from our papers [Jotsov1 1999, Jotsov2 1999]. Compared with the works of Lenat [Lenat 1983], or with other sources in the references on informatics, the majority of our papers describe the mathematical results, not the method. The paper's scope is *interdisciplinary* and includes many significantly far research areas. To some extent the proposed method is a continuation of the Lenat's ideas and serves the same **purposes**: elicitation of new knowledge in the integer data processing, derivation of new formulas, and *whenever possible* generation of new mathematical theorems. At the same time it has some points in common with the Narin'yani's, Shvetsov's constraint programming [Narin'yani 2000,Shvetsov 1997] and reasoning in the Altshuller or Hadamard or Polya style [Altshuller 1979, Hadamard 1975, Polya 1963]. The approach is enriched from the most remote principles coming from both directions but it *uses no plausible reasoning*.

2. The FRONTAL Method and the Working Environment

The shortly described below FRONTAL method interacts with several other methods under the common control of a new type of an evolutionary metamethod. The metamethod avoids or *defeats* crossovers, phenotypes, mutations, etc. Below we choose the description in an analogous manner as the way to reduce the extra descriptions, because the general scheme of the chosen strategy is rather voluminous. The evolutionary metamethod swallows and controls the following methods:

I. FRONTAL method;
II. KALEIDOSCOPE method;
III. FUNNEL method;
IV. CROSSWORD method.

The KALEIDOSCOPE method is the *background* for the human-machine strategies for work. The machine forms and visualizes different mappings for the chosen groups of numbers or like, while the obtained results are estimated by the human. *The human* makes the necessary conclusions and undertakes the required steps. Analogically the kaleidoscope rotations form different images in a hazardous manner, and the spectator takes an *informal decision* whether the seen by him is nice, original etc.

Let's assume you have a *plastic funnel*. If you fix it vertically above the ground, you can direct a stream of water or of vaporous drops etc. If you change the funnel direction, then the stream targeting will be hampered. Fixing the funnel horizontally makes it practically useless. Analogically in the evolutionary method the general direction in numerical models is determined likewise. In other words this is a movement along the predefined gradient of the information. This term is proposed in a manner which *has some connection* to [Baldi 1995]. Just like in the case of the physical example in the beginning of the investigation there are lots of undirected hazardous steps towards conclusions and hypotheses. The FUNNEL method is based on inconsistency tests with known information.

Let us assume that the reader solves a problem with a complex sentence of 400 letters with vague for the reader explanations. Let the unknown sentence be horizontally located. The reader can't solve the problem in an arbitrary manner, because the number of combinations is increased exponentially. Now it is convenient to facilitate the solution by linking the well known to the reader information with the complex one from the same model. The reader tries to find vertical words that he is conscious about like the place of our conference KDS 2003 - Varna. The more the crosspoints are, the easier is the solution of the horizontal sentence. The approach for the CROSSWORD is *even easier*. Here both the easy meanings and the difficult ones are from one domain, therefore there exists an additional help to find the final solution.

For pity the paper length does not allow us to make more detailed descriptions of the mentioned above methods, and/or their connections, interactions, etc. We will turn exclusively to the considered FRONTAL method.

The trend in the investigation includes solutions of complex hypotheses and problems which require the usage of integer-number models. Great number of these problems have been unsolved for centuries; their decisions cannot be obtained *prima vista* or in a frontal manner. This is the reason for the development and application in mathematics of an evolutionary strategy. In it the preproofs are on the first place. In the process of solving oversophisticated problems the first draft solutions comprise only the first step in the marked by the FUNNEL direction. This direction is an approximate. This is due to the initial conditions and knowledge constraints. Fig. 1 depicts a similar general direction for research by the A-B line. The obtained intermediate solutions follow another route, A-C-D-B. The solution B is inaccessible from the node C or from any other node before D. The user can change the direction according to her/his wish. D-E on Fig. 1 is a deviation from the line A-B. The new branch marks the process of solving another problem. Any of the intermediate solutions may contradict or doesn't correspond to the final solution (B). Together they form the set of preproofs for B. The mathematical proofs are formed in the process of evolution with no probabilities. In the evolutionary metamethod the preproofs are usually weak, with bottlenecks and/or incomplete. The preproofs in the considered domain are never so good as to be included in the "official" proof. Nevertheless they must not be easily rejected. They are weaker, but in our case they *are not* heuristical by nature, and they might assist the solution of other problems as well.

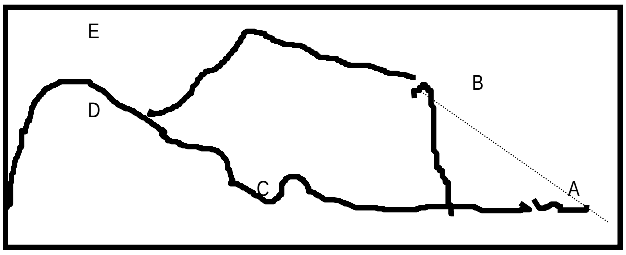

Fig. 1.

The presented evolutionary meta-method has the following features. The solution is evolved *step by step*. At every step it is possible to have a progress or a regress compared with the previous decision. The role of probabilities and other subjective estimations is played by interactive approaches for knowledge acquisition, data linkage, mappings and other processing of data and knowledge. The investigated FRONTAL method (I) includes the following procedures. Their short abbreviations are given in bold letters.

1. MOC: Mix Or Change (data/knowledge);
2. BIND: Connects the information (data sets/knowledge) during the automatic work or shows it to the user;
3. WHY & HOW: Forwards it (data sets/knowledge) to the user;
4. CS: Constraint Satisfaction (of knowledge), based on the weak negation ~;
5. SPREAD (knowledge);
6. WHAT: Explanation (of data/knowledge);
7. EF: Elimination Filter.

All the seven procedures can be modified together with the change of the different models. Now we introduce in short the FRONTAL method terminology. Let M be a set of such models M_i which contain sets of arithmetic progressions $\{a_i+b_ik\}^{\infty}_{k=0}$. At that:

(1) $$b_i = \prod_{p_j \in M} p_j ; \quad p_j \in P.$$

where P denotes the prime numbers set. Every progression from M_i may be treated as a result after sieving out the set of positive integers, consisting of all p^i_M and such composite numbers that at least one of p^i_M divides them. To simplify the contents other models are not included, e.g. based on geometrical progressions. It is accepted that $(a_i,b_i)=1$; $a_i<b_i$.

Four operations are introduced in every model: {+,-,*,/}. Possibly every application of the algorithms based on the FRONTAL method leads to some change of different parameters inside the built-up algorithms whenever the model changes. This model changes serve as *an algorithm stability test*. This is the right place to use MOC. Denote $V=\{v_1,v_2,...v_z\}$ is a set of parameters. During our first investigations in the eighties we used V in a way similar to the genotype from Genetic Algorithms (GA). The user had the option to accept such v_i which deserved his attention and the system proceeded with the goal task. We offered that every task must begin with $V=\{\varnothing\}$. Thus the *released assumption* brings the user closer to data mining tasks.

The author proposes the following generalized MOC algorithm with an automatic mode set-up: A. Fixing of v_i in the current model; B. Case-based inclusion of v_i from previous solutions; C. The algorithm proceeds with review of $v_i=0$; D. An inverse mapping of (C.) is introduced or $v_i\to max$; E. v_i is replaced by another parameter in V; F. The algorithm goes on with the WHAT procedure or with other procedures from the FORWARD method. The general MOC scheme is postulated with the formulas (2) and (3).

(2) $$S(V\to V'); \ card(V)\neq card(V').$$

(3) $$L(S(v_{i,k}))\to L(S(v_{i,j})); \ S(v_{i,k})\neq S(v_{i,j}).$$

Here S is a situation which has arisen as a result from the MOC activity changing the set V or its separate element v_i. L is the modal operator *possibility*.

For example, let v_5=2 means that all the numerical data are copied in a bidimensional array. This automatically inputs $v_6 = \vec{x}$ and $v_7 = \vec{y}$ in V. During the activation of (D.) $v_8 = \vec{z}$ is introduced, etc. When processing (C.), the bounded with v_5 parameters v_6=0 or v_7=0 are affected. In this way MOC acquires new knowledge from the data investigation. The next example is not so theoretical. Rather it is connected with numbers from eight arithmetic progressions.

The following denotations are introduced. $\{m+nk\}^{\infty}_{k=0}$ is an arithmetic progression (progression for short). In it m is the first member, and n is the step. $\pi(x)$ is the total number of the primes which are elements of the set P ($p_i \in P$, $p_i \le x$). $\pi_{n,m}(x)$ is the number of primes $\le x$ which are contained in the progression. S_5 is an union of 8 progressions $\{y+nk\}^{\infty}_{k=0}$, $y \in Y$, Y={1,7,11,13,17,19,23,29}. Every of these progressions is represented as a column in Fig. 2 if the elements of S_5 are shown vertically. Fig. 3 shows the same environment in a slightly different manner. Every of the elements in S_5 is computed in the following way. The first number from the corresponding column - see line 1 - is added to the number from the same line and the leftmost column. For example $s_{14,2}$=7+390 is in line 14 and column 2. Composite numbers in S_5 are represented as products of prime numbers. The primes are the result of the decomposition of the composites. In Fig. 3 the primes are *omitted* while the particular cases $y \in Y$ are given in brackets. MOC has no logical inference. It simply finds and changes the scope parameters one by one while the rest of the parameters remain unchanged. The lines below show the cases when MOC pastes or cuts some of the elements in the interpretation. For example during the investigation of the operations addition and multiplication in S_5 the following parameters attract the attention: primes (with just a single divisor), composites with at least 2 divisors, 8 columns which are parallel to the vertical axis $\vec{y}$ and 15 lines which are parallel to $\vec{x}$. These 4 parameters can have other designations, which will have similar meanings. The names are not significant. The parameters are established by mere observations e.g. directly on the figures. The following transforms for the transition from Fig. 2 to Fig. 3 are used:

(T_1). The primes are determined but not shown from all the numbers in the fragment, see Fig. 2. The *very omission* introduces some new information. The figures below demonstrate the following versions of transformations in S_5.

(T_2). All the composites are presented as products of prime divisors.

(T_3). All the composites with the divisor of 13 are successively connected with straight lines.

1	7	11	13	17	19	23	29
31	37	41	43	47	49	53	59
61	67	71	73	77	79	83	89
91	97	101	103	107	109	113	119
121	127	131	133	137	139	143	149
151	157	161	163	167	169	173	179
181	187	191	193	197	199	203	209
211	217	219	223	227	229	233	239
241	247	251	253	257	259	263	269
271	277	281	283	287	289	293	299
301	307	311	313	317	319	323	329
331	337	341	343	347	349	353	359
361	367	371	373	377	379	383	389
391	397	401	403	407	409	413	419
421	427	431	433	437	439	443	449

Fig. 2.

(T_4). All the composites with the divisor of 17 are successively connected with straight lines. The result is shown in Fig. 4. The transformation itself is in the divisor replacement.

(T_5). Besides the graphical interpretations in Fig 3 and Fig. 4 must be added similar pictures for the “neighbors below” 43 and 47 or 13+30, 17+30. The result has the same succession of beat for the columns with periods 30 times 43 and 30 times 47. The illustrations resemble the Fig. 3 and Fig. 4 but they are more elongated due to the greater period.

(T_6). The parameter influence of $\vec{x}$ is "reduced". So the attention is concentrated upon the **beat succession** for the columns S_5 and the lines are "compressed". The results are depicted in Fig. 5 and Fig. 6.

0	(1)	(7)	(11)	(13)	(17)	(19)	(23)	(29)
30						7^2		
60					$7\cdot 11$			
90	$7\cdot 13$							$7\cdot 17$
120	11^2			$7\cdot 19$			$11\cdot 13$	
150			$7\cdot 23$			13^2		
180		$11\cdot 17$					$7\cdot 29$	$11\cdot 19$
210		$7\cdot 31$	$13\cdot 17$					
240		$13\cdot 19$		$11\cdot 23$		$7\cdot 37$		
270					$7\cdot 41$	17^2		$13\cdot 23$
300	$7\cdot 43$					$11\cdot 29$	$17\cdot 19$	$7\cdot 47$
330			$11\cdot 31$	7^3				
360	19^2		$7\cdot 53$		$13\cdot 29$			

Fig. 3.

0	(1)	(7)	(11)	(13)	(17)	(19)	(23)	(29)
30						7^2		
60					$7\cdot 11$			
90	$7\cdot 13$							$7\cdot 17$
120	11^2			$7\cdot 19$			$11\cdot 13$	
150			$7\cdot 23$			13^2		
180		$11\cdot 17$					$7\cdot 29$	$11\cdot 19$
210		$7\cdot 31$	$13\cdot 17$					
240		$13\cdot 19$		$11\cdot 23$		$7\cdot 37$		
270					$7\cdot 41$	17^2		$13\cdot 23$
300	$7\cdot 43$					$11\cdot 29$	$17\cdot 19$	$7\cdot 47$
330			$11\cdot 31$	7^3				
360	19^2		$7\cdot 53$		$13\cdot 29$			
390	$17\cdot 23$			$13\cdot 31$	$11\cdot 37$		$7\cdot 59$	
420		$7\cdot 61$			$19\cdot 23$			
450	$11\cdot 41$					$7\cdot 67$	$11\cdot 43$	
480	$13\cdot 37$			$17\cdot 29$	$7\cdot 71$			

Fig. 4.

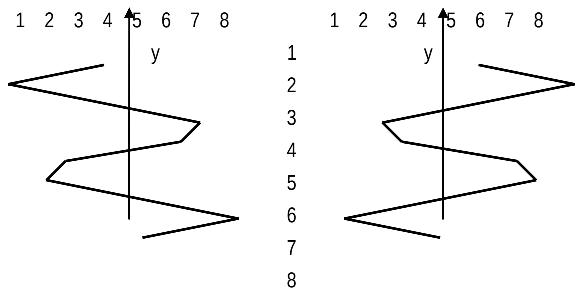

Fig. 5. Fig. 6.

The discussed six relatively simple transformations show plainly and unambiguously that the cited in Fig. 5 way to beat the columns is one and the same for all the elements in column 4 in S_5: 13, 43... The result is in relation with the transition from a piece of S_5 to the whole S_5 or v.v. It is specially discussed in the SPREAD presentation. Fig. 6 presents the situation with the elements in column 5 (17,47...) which is analogous.

The two numerical sets in Fig. 5 and in Fig. 6 interpret the same cycles as those in Fig. 3 and Fig. 4. These cycles have common "similarity centers" on $\vec{y}$. Moreover the two figures coincide if one of them is rotated 180 degrees around $\vec{y}$ (T_7). The revealed dependency is valid only for numbers of the type n and 30k-n for every positive integer k. If the beat cycle for the columns in Fig. 5 is in a column starting with the element m, then the analogical cycle in Fig. 6 is in a column starting with 30-m. The constantly repeated number 30 leads to (T_8): 30=2·3·5. *The act of mathematical creation* for Fig. 2-Fig. 3 is unambiguously simple when mapping Fig. 5 to Fig. 6. The revealing of different numerical properties takes place in the described above MOC procedure. Other transformations can be pointed like (T_9): the discovery of numbers which *can't be divisors* of any integer number. Zero which is not an element in S_5, but being a similarity center for the positive and negative parts in S_5, is set in this manner. The interpretation of any prime cycle as on Fig. 7 is unified by the total discrimination of the influence of $\vec{y}$; (T_{10}) is a suitable example as an illustration vs. (T_7).

1 2 3 4 5 6 7 8

• • • • • • • • • 1

Fig. 7.

All discussed transformations are just consequences of observations based on the model. They give no answers to questions like WHY or HOW the presented results are obtained. The body of the preproofs is formed on the basis of such conclusions.

The achieved with MOC results may be related and compared. This is the purpose of the BIND procedure. The extracted information is analyzed by BIND on the basis of juxtapositions. BIND is based on the above function mapping $S_{x1}(v_1,...v_z)=S_{x2}(v_1,...v_z)$ or $S_{x1}(v_1,...v_z)\neq S_{x2}(v_1,...v_z)$ where x_i are different objects or data groups. The detailed BIND overview exceeds consideration line in the paper. The obtained results most of all lack of proving power and the inference obtained is nonmonotonous. Therefore after determining the regularities it is possible to formulate prompting queries to the user which are decorated in the well known form WHY and HOW. The system forms the basis for the general solution, and the details are an object for a manual or an interactive work. In this way, the investigation evolves itself. Using the WHY&HOW procedure, a new set is built from mutually related formulas and knowledge from the same domain.

The CS procedure is formalized in a manner similar to the one in [Narin'yani 2000]. An outstanding feature of the presented variant of CS is that the bounds of the domain are not restrictive in the case of a weak negation ~. After the contradictory resolution these bounds are overcome. The contradiction concentrates the attention to the incompleteness in the scope. The goal-forming scenario in the constraint satisfaction paradigm is formulated as follows. Let the variables x_1, x_2, ... x_n be the mapped sets of their value spaces X_1, X_2, ... X_n. The constraints $C_j(x_1,x_2,...,x_n)$, j=1,...k are valid for the same X_j. It is necessary to find such sets $\langle a_1, a_2, ... a_n\rangle$ such that $a_i \in X_i$ and they satisfy all C_j simultaneously.

Denote M* is a *subdefinite model* or - roughly speaking - an incomplete model. Let $C_i(x_1,x_2,...,x_n)$ is one from the investigated constraints, and $N(x_1,x_2,...,x_n)$ be such that:

(4) $$N(x_1,x_2,...,x_n) \rightarrow \sim C_i(x_1,x_2,...,x_n).$$

This means that the constraint is violated because (4) contains the weak nonclassical negation ~. The ~ based inconsistencies may be solved after the complementation of M* with new knowledge/data. The augmented model is denoted with M'. In it the examined constraint takes the form $C'_i(x_1,x_2,...,x_n)$, z=n or z≠n, where:

(5) $$C'_i(x_1,x_2,...,x_n) \rightarrow \sim C_i(x_1,x_2,...,x_n).$$

There are other possible ways for the transition M*-M' besides $N(x_1,x_2,...,x_n)$. One of them is to include a new parameter v' in M*. Another approach is possible in the case when the system of constraints has no solution. Often in such cases there exists an information which admits the re-examination of $C_i(x_1,x_2,...,x_n)$. For example, let us examine the numbers $x \geq 11$. Then we may come to the conclusion that:

(6) $$\pi(x) > \frac{x}{\ln(x)}.$$

Here M* has no constraints and $C_i=\{\emptyset\}$. The result can be monotonously generalized to the whole interval $[0,\infty]$. The case when x=8 violates the formula (6). This contradicts the assumptions especially the case $C_i=\{\emptyset\}$. The introduction of C'_1: $x \geq 11$ leads to the result:

(7) $$\pi(x) \underset{x \geq 11}{>} \frac{x}{\ln(x)}.$$

The last three procedures do not contain substantially new theoretical ideas. SPREAD is based on the well known concept of mathematical induction. WHAT is designed to communicate with humans, because the internal representation of the solutions is obscure. EP serves as a barrier against knowledge duplications or a surplus knowledge.

The interaction between the first five procedures is on a competitive basis according to the JUNGLE principle. In some cases they act in the role of *demons*. In the rest of the cases the top priority is assigned to the procedure from the previous iteration or this one which has generated the most effective solutions. The following formalization is aimed to derive this simple estimates and agreements. JUNGLE is based upon estimates $0 \leq f(Q_i) \leq 1$ for every procedure of the FRONTAL-based set $Q=\{Q_1,...,Q_7\}$. In this case it is preferable to compare the described JUNGLE strategy with the one from GA "the fittest wins" ([EAEA 1997], p.3). We use it in the form "the winner is best estimated". If $f(Q_i)=1$, then the procedure interacts with EF and the user. If $0.25 \leq f(Q_i)<1$, then the display contains this value, and the corresponding solutions are considered only on the user request. The user may interfere in the automatic process of the estimation. The threshold value $f(Q_i)=1$ is achieved in the following situations:

(8) $S(Q_j) \rightarrow G(Q_i)$; $j \neq i$; $i,j=1...7$; $G(Q_i) \rightarrow f(Q_i)=1$.

where G is the modal operator *necessity*, $S(Q_j)$ is a scenario in Q_j leading to $G(Q_i)$. An example of (8) is presented above after (T_6) thus activating SPREAD by MOC.

(9) $U \rightarrow f(Q_i)=1$.

Here U means *user-defined activation*. The user defines the necessary parameters for Q_i.

(10) $S(Q_3)=c \rightarrow f(Q_i)=1$.

where $S(Q_3)$ is the BIND output. The meaning of c (for short from convergence) is that *the results from the two independent research lines coincide*. Fig. 7 depicts an example leading to $S(Q_3)=c$. In the future JUNGLE may incorporate Machine Learning (ML) approaches. At that:

(11) $S(Q_3)=a \rightarrow G(Q_i)$.

(12) $S(Q_3)=e \rightarrow G(Q_i)$.

where a means "the memorized logical inference is abbreviated"; e means *an explanation of the obtained earlier results*. $f(Q_i)<1$ is obtained in the following cases:

(13) $f^p(Q_i)=\max_j(f^p{}_j(Q_i)), j=1,...7 \rightarrow f(Q_i)=0.5\, f^p(Q_i)$.

where $f^p(Q_i)$ are all the memorized evaluations in MOC.

(14) $f^p(Q_i)=\max_t(f^p{}_j(Q_i,t)), j=1,...7 \rightarrow f(Q_i)=0.7\, f^p(Q_i)$.

Only the last remembered value for the corresponding $f(Q_i)$ is taken into account in (14). Some of the above presented procedures are included not only in the FRONTAL, but also in the neighboring methods. The set of all those methods uses the same JUNGLE principle.

The goal function *is easy to change* (see Fig. 1), so the procedures from 1 up to 7 may operate not only with data, but also with goals. E.g. BIND can operate with hypothesis I with hypothesis J in S_5, etc.

3. Experimental Studies and Some of Theoretical Results

The software for the research includes more than 20 programs written in Visual Basic and more than 200 MB Excel data. The assistant and defensive software consists of more than 20 programs in C and C++.

The introduced method generated new results even during the first investigations in 1986. The following strategy was formulated later. The target is to find dependencies in the arrangements of different sets of numbers, e.g. which are multiples of 17. (For example see Fig. 4 and the multiplication cycle 17). One can say that the start is with *zero information*. We introduce descriptions of well known hypotheses, e.g. the twin primes hypothesis, Goldbach's conjecture etc. in the same model. Finally we obtain new mathematical dependencies and formulas. In practice this approach starts with a research of the twin primes hypothesis with a difference of 2: these are couples of prime numbers 5 and 7, 11 and 13 etc. The hypothesis is based on the suggestion that there exist an infinite number of such similar pairs. The hypothesis formalization *must not be mistaken* with the goal function. It is simply a model inside the given sets of progressions. The research of the multiplication operations with prime numbers in different numerical models, e.g. in S_5 leads to the conclusion that the principle properties of different composite numerical unions are also prime number functions (15), (16)! This result at a first glance is very remote from the twin primes hypothesis. This result relates to the proof of Theorem 1 which was not a target in the research. Nevertheless it may assist in the process of solving for many different goals. The famous Dirichlet's theorem is a corollary from the Theorem 1.

$$c_{K,6,1}(x) = \sum_{p=7}^{p_{z7}} c_{K-1,6,1}\left(\frac{x}{p}\right) + \sum_{p=5}^{p_{z5}} c_{K-1,6,5}\left(\frac{x}{p}\right). \tag{15}$$

$$c_{K,6,5}(x) = \sum_{p=5}^{p_{z5}} c_{K-1,6,1}\left(\frac{x}{p}\right) + \sum_{p=7}^{p_{z7}} c_{K-1,6,5}\left(\frac{x}{p}\right). \tag{16}$$

where $p_{za} \in \{a+6k\}^{\infty}_{k=0}$, $C_{k,6,a}(x)$ are all the composites $\leq x$ from $\{a+6k\}^{\infty}_{k=0}$ which contain k prime divisors.

Theorem 1.

We have the interval [0,x]. In it we have two progressions $\{m_1+nk\}^{\infty}_{k=0}$ and $\{m_2+nk\}^{\infty}_{k=0}$ and the relevant numbers are mutually prime: $(m_1,n)=1$, $(m_2,n)=1$. Denote $\Delta\pi_{n,mi}(x)$, i=1,2. The denotation introduces the difference (delta) in the number of the primes $\leq x$ included in both progressions. This difference may not be greater then the number of the primes in the range $[0, \sqrt{x}]$, which is signed as follows: $\Delta\pi_{n,mi}(x) \leq \pi(\sqrt{x})$.

The Theorem 1 proof is given in [Jotsov2 1999]. Theorem 1 is the basic tool for the derivation of the twin primes formula:

$$P_X(p, p+2) \sim 1.320323632 \frac{(\pi(x))^2}{x}. \tag{17}$$

where $P_x(p,p+2)$ is the number of twin prime couples $\leq x$, ~ means "asymptotically equal". The solutions below are related to the well known Hardy-Littlewood's hypothesis, the formalization of which is introduced in (19). The formalization check of it revealed a series of inconsistencies, so the hypothesis was transformed in (18). Finally the FRONTAL method has lead to a new hypothesis 1 which is stronger than the Hardy-Littlewood's.

$$P_X(p, p+d_1, \ldots p+d_{z-1}) \geq K_z \frac{(\pi(x))^z}{x^{z-1}}. \tag{18}$$

where P_x is the number of z-tuples $\leq x$. They have different admittable differences between, and K_z are the corresponding coefficients [Riesel 1985].

$$P_X(p, p+d_1, \ldots p+d_{z-1}) \sim K_z \frac{x}{(\ln x)^z}. \tag{19}$$

Hypothesis 1.

Denote z the arithmetic progressions $\{a_1+b_1k\}^{\infty}_{k=0}$ ... $\{a_z+b_zk\}^{\infty}_{k=0}$ with a *noncoinciding* step of progressions. Let all the corresponding (a,b)=1. If z-tuples of positive integers $(c_i,d_i,\ldots z_i)$ are compared; all of them are positive integer numbers; $c_i=a_2-a_1+(b_2-b_1)(i-1+w_1)$ ... $z_i-c_i=a_z-a_1+(b_z-b_1)(i-1+w_z)$, and w are positive integers, then there exist infinitely many such z-tuples $(c_i, d_i, \ldots z_i)$ in which all the numbers are primes $c_i \in P$, $d_i \in P$, ... $z_i \in P$.

Hypothesis 1 is formulated as a result of the application of Theorem 1 to the formula (18). Finally the MOC procedure was applied to the model of the Hardy-Littlewood's hypothesis in S_5. At the end we shall reveal an indicative fact. The paper containing the draft with the Theorem 1 proof is one page long. The initial version of the theorem comprised more than 30 pages with several bottlenecks. The author improved the proof using manually the FRONTAL method and the CROSSWORD method. The obtained by now results confirm the effect in cases with *infinite* sets of integers and they reveal possibilities for solving problems with *higher complexity*.

4. Some of the Advantages

The greater part of the seven procedures and their interaction inside the FRONTAL method are completely original. This method operates in the *environment of other methods* which are also proposed by the same author. The usage of this method in Number Theory leads to new mathematical results which are widely discussed and acknowledged as original. Part of them is accepted for a publication in Australia. Another fraction is under consideration in AMS. The results from section 3 after Theorem 1 are only partially issued in the math periodicals. They are presented as an illustration of the method for the way in which a front of mutually related solutions can be formed. It is possible to set a way for applications of contemporary IT in computational mathematics, residing on the presented method.

5. Conclusions

A new IT method is proposed for the interactive construction of formulas and proofs in Number Theory. It follows from the consideration that *even a non-specialist* can make easy explainable solutions if she/he uses the present work with the described method. The method is multi-target oriented and its main part is domain independent.

Bibliography

[Altshuller 1979] G. S. Altshuller, *Creation as an Exact Science*, Moscow, Sov. Radio, 1979.

[Baldi 1995] P. Baldi, "Gradient learning algorithm overview: A general dynamical systems perspective," *IEEE Trans. on Neural Networks*, Vol. 6, pp. 182-195, Jan. 1995.

[EAEA 1997] *Evolutionary Algorithms in Engineering Applications*, D. Dasqupta and Z. Michalewicz (Eds.), Springer, Berlin etc., 1997.

[EA 1997] *Evolutionary Algorithms*, L. D. Davis et al. (Eds.), Springer - Verlag, New York etc., 1999.

[Hadamard 1975] J. Hadamard, Essai sue la Psychologie de l'Invention dans le Domaine Mathematique, Paris, Gautier-Villas, 1975.

[Jotsov1 1999] V. Jotsov, "On one approach in proof search and refinement," *J. Artif. Intell*. (National Academy of Sciences - Ukraine), No. 2, pp. 97-103, 1999.

[Jotsov2 1999] V. Jotsov, "On the usage of discovering systems' approaches in Number Theory," *IIT Working Paper* WP/83B, Bulg. Acad. of Sciences, 14pp., December 1999.

[Lenat 1976] B. Lenat, "AM: an artificial intelligence approach to discovery in mathematics as heuristic search," *Memo* AIM-286, Stanford University, CA, 1976.

[Lenat 1983] D. B. Lenat, "Why AM and EURISKO appear to work," *Artif. Intell*., Vol. 21, pp. 61-98, 1983.

[Narin'yani 2000] A. S. Narin'yani et. al., "Constraint programming and subdefinite models," preliminary version.

[Nordin 1999] P.Nordin and A. Eriksson, M. Nordahl, "Genetic reasoning: evolutionary induction of mathematical proofs," in R. Poli, P. Nordin, W. B. Langton, T. C. Fogarty (Eds.), Genetic Programming, *LNCS* 1598, Springer, Berlin etc., pp. 221-231, 1999.

[Polya 1963] G. Polya, *How to Solve It*, Princeton University Press, N.J., 1963.

[Riesel 1985] H. Riesel, Prime Numbers and Computer Methods for Factorization, Birkhauser, Boston, 1985.

[Shvetsov 1997] I. E.Shvetsov, V. V. Telerman, and D. M. Ushakov, "NeMo+: object-oriented constraint programming environment based on subdefinite models," Proc. Third Intern. Conf. on Principles and Practice of Constraint Programming (CP'97), *LNCS* 1330, Springer, Berlin etc., pp. 373-385, 1997.

Author information

Vladimir S. Jotsov – College of Library Science Education and Information Technologies (KBIT); Intstitute of Information Technologies – Bulgarian Academy of Sciences (IIT-BAS); 1113 Sofia, P.O. Box 161, Bulgaria; tel. 359 2 9708592, e-mail: jotsov@ieee.org