Highly Successful Models In The Applied Mathematics Teaching

Oleg I.Krivosheev
Lab 40: Laboratory of Intellectual Control Systems
Institute of Control Sciences RAS
Moscow, Russia
o-krivosheev@ya.ru

***Abstract*—This article is devoted to an obvious terra-incognita in contemporary education and Atlántida on the natural science planet – the applied mathematics subject. This subject normally should be the central one after the practice-orientation principles (likely wrong enough) were announced as the main educational keystone, but it isn't. The strange negative success of this discipline can be explained partially by its' specific (long chain methods) and partially by not understanding this circumstance. In the article, we continue the chain of publications explaining how one should build an applied mathematics course based on malty-parametrical problems [2]. Here we talk about applied mathematics course possibilities and content constraints.**

Keywords— applied mathematics, parametrical problems, course construction, algorithmic problems learning, algorithmic methods learning difficulties, applied mathematics learning.

I. Why Do Applied Mathematics Nearly Ignored In The Study Programs

In contemporary education, there is a huge gap: in a good case we have fundamental mathematics (of different levels) and some very specific applications, but we always (or nearly always) do not have a standard bridge from fundamental (or basic) mathematics (like linear algebra or square equation solution) to wide-spread problems. This is due to some objective circumstances. The first one is that the applied mathematics problems are much longer than most fundamental methods (the possibility and success of learning fundamental methods in any given student group do not mean the possibility of applied math methods learning) and second – that would not be so bad if we had enough choice of applied methods, but in fact, the situation is different to one we have at a secondary school disciplines like physics, chemistry, and mathematics, where we have very wide combinatorics of different study problems. In the general applied mathematics for manager, finance, and economics specializations (not for engineers) the author had found hardly more than 50-70 highly successful unique algorithms and problems and hardly more than 30-40 highly successful laboratory works for non-linearity and neural network construction (we don't permit any laboratory, that can't be done in two pairs from a white sheet – no black boxes are allowed). And even if we look at a wider problem set of general algorithmicizing (and encoding) studies the successful problem number remains almost the same – definitely less than 200 and in fact enough less than 100.

So, we have a very thin set of algorithms (models and methods) spread in very different fields from decision-making to game theory, from standard operation research to discrete math and algorithmizing.

These methods at one time require both high qualifications of a lecturer and students or a teacher and pupils. Moreover, nearly always it also requires much more time per one example if compared with fundamental subjects.

That's why the standard solution of administrations is to exclude the applied mathematics courses, except for the very seldom situation when top-qualified teachers met with uniformly highly-qualified pupils (or students). And the hardest requirement is often uniformity of the high students' level in the group. The probability of it is low (if we have a heterogeneous mixture of commercial and ordinary students – very low) and administrations are faced with a bad choice: to expel the low-level commercial or budget students (losing money) or to make some courses non-obligatory with a corresponding dramatic decrease in the final study quality.

II. Task-Based Learning

As soon as the Moscow Institute of Physics and Technology was organized in 1946 near Moscow, it aroused a problem. Too many students (with very good, maybe brilliant, initial quality) damaged their health (even coming to the mad house). The reason was that it was one of 3-4 Soviet universities with a very hard program. In this case, it was due to the mechanical unification of physics and natural sciences in extended MSU Physical faculty amount from one side and mathematics in standard MSU Mathematical Mechanical faculty volumes from another, used in the very beginning. Later there was found another system (called the Phys-Tech system), made it possible to learn approximately the same extended area at appropriate difficulty levels and time expenses based on problem-solving. Still, we can't call it 'problem-based learning' since this notion is already wide-used in slightly different senses and that difference is highly correlated with at least Russian-language tradition when a task means an ordinary problem (maybe well-formalized) and the Problem itself means a substantially huge problem, that possibly has no solution in any sense and current interpretation of problem-based learning more or less corresponds to the last variant starting from the moment it was introduced for medical-college students [4] and further applied in different cases (including highly formalized courses) like [5]. In fact, it's the other side of the medal – for a poorly formalized area we are forced to use poorly formalized tasks requiring long group discussion and co-working. In the consequent paragraphs we mean the same practice except for very well formalized tasks, able to be solved in a classroom when the malty-variant [2] abcd-parametric technique permits to make each solution a useful example without any cheating possibility – as soon as calculations stay strictly individual.

III. Active Learning In Application To Algorithmic Methods Specifics

Despite all the difficulties connected with the long methods of learning, a successful course in applied mathematics is possible - even if we deal with not very highly

979-8-3503-2656-7/23/$31.00 ©2023 IEEE

qualified students and either with strong time limits for a teacher.

Still, it requires proper methodology [2], based on manual analog of [6,7] (automated problem generation), keen method complexity control, and teaching sequence management. Since long methods are more like programming, than classical problem-solving. So we should permit and force a student to run the theoretical algorithm at a pencil and paper variant. It may be done at home (the harder variant permitting 30-55% efficiency at a longer time) or in the classroom when all the students solve similar but numerically different problems from the same 4-parametrical set like [1]. The problem of long-methods learning is that they comparatively exceed normal student operative memory possibilities, while simultaneous solution allows one to understand a method or an algorithm after it is implemented in the numeric paper & pencil solution. That reduces the necessary time by 50-80% (2-5 times) and permits efficiency up to 80%.

IV. Information Transmission Problem And Motivation Necessity

As soon as multimedia projectors were widely introduced in the educational process there appeared a problem when we upgraded teacher as a transmitter but we had the same pupils (students) receiving capacities.

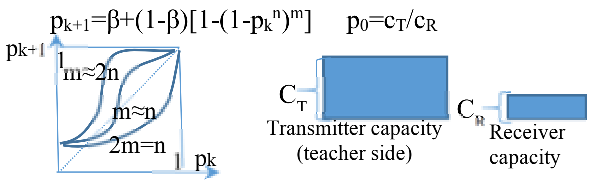

Fig. 1 Learning Kapustin Model. The multimedia projector and the receiving ability of the student produce initial catching probability p_0 for an algorithm step of n-step algorithm. That changes $p_{k+1}=\beta+(1-\beta)\ [1-(1-p_k^n)^m\ ]$ in time k, where m - a number of chains, a pupil or student can build a solution using his neighbors' understanding. It is dependent on the density of active pupils or students in the auditorium. Starting from a rather low position p_0 (equal to the ratio of transmitted and received (depending on operative memory) data). Here β – is external, mainly teacher's, help either bounded by teachers and course time resources.

As soon as we overcome the student's receiving part constraint by introducing an integrated seminar technique (overcoming student's operative memory constraint), which is nearly impossible without most of student's active position we come to the necessity of student motivation almost like a motivation of a soldier. We face here an individual and collective psychology problem, when the first – most active student or soldier estimates the possibility of action and the others follow him trying to reproduce his results somebody 50%, somebody more or less, depending on that information spreads like in the [2] peering net, when reliability theory (Kapustin model) formulas act [1, pp.154-157]. In that case, Grannovetter (Izing) type self-interaction of the system makes it potentially bi-stable, when active equilibrium is not a default option and should be accurately held by following winning trajectory. That will force us to follow the consequent complexity-based timetable-building algorithm (first we treat with ice-cream, then fruits, then snakes and frogs).

V. Discrete or Continuous Course

Since we need complexity control, further we consider discrete problems only. This excludes a huge part of continuous media problems (like [3]) typical for engineering courses. That does not necessarily mean excluding corresponding methods: the enormously difficult continuous Pontryagin maximum principle may be explained at first or at one of the first lections (or integrated seminars) in its' primitive discrete variant. Still, we do really omit everything in touch with partial differential equations, and continuous probabilistic distribution and limit ordinary differential equations to calculational laboratory work part (with no analytical parts).

VI. Successful Methods

If we would look into the structure of learning of nearly any non-humanitarian subject we will see, that many of the contemporary worth talking about ideas are often marked with Nobel Prizes (especially in physics, economics, etc.), all the old ones are not marked with Nobel Prizes seemingly only due to the death of the inventor before the prize established. So, when we talk about a successful algorithm, model, or applied method we mean that it should be something like a typical Nobel-prize result – at least at the level or considerably better (as sometimes Nobel results seem strange and politically correct).

Some ideas of non-engineering applied mathematics methods and problems were an epoch in management (like storage control, net-project management) forming a main-stream at least for a decade, some were not, but were applicable to a very general set of situations like spanning-tree, COMM voyager, mass-service or linear programming problem set.

At purely popularity criteria we should recognize these methods as often more valuable than many ones officially marked.

VII. Can We Make Applied Mathematics Courses Specific For Students Of Different Non-Engineering Specializations

Our answer is no, we cannot. Until the scarcity of good illustrative examples is overcome, we should train brains in almost one and the same training set. That does not deny introducing more problems in some specifics but the core one should keep untouched. This is very obvious when getting acquainted with advanced courses in discrete mathematics [10], and classical (narrow) operation research – they are nearly never developed in-depth, but usually extensively regard neighboring subject themes.

In most of the subject areas, we do not have more than 10 (at least 14-15) successful models, algorithms, or methods.

VIII. The Complexity Arrangement Metodics

Since complexity is our main problem, now we should estimate the complexity of the methods. It permits us to make a proper sequence in the future.

Now we have listed core methods worth mentioning in a universal course for non-engineers. It is about 40 theoretical and almost the same number of practical methods and study problems (or computer laboratory works for C# and Excel sheet).

As the author has taught about 6000-7000 students in more than a hundred groups and group flows [2]. One time there even observed an untypical situation when nearly each hundredth current student in Moscow had earlier studied this course or its lighter (and earlier) variants - so, the author has got a great experience of how one definitely shouldn't build a course and how one should build a stimulation policy.

At first, we used very specific melodics optimized for long operation chain methods [2] that are different from the short ones the students (or pupils) face in fundamental courses. We plasticized lection practical work and seminar mixture totally based on huge variant set parametrical problems (4 parameters gave us nearly 10 000 variants, that excluded copy–past mind-free write-off instead of knowledge exchange and spreading). This method has no alternative to the fastness of long-chain method learning but the students (pupils) should have time to get in touch with. As we used brigade – collective study method (see [2]), all the group members needed to get the right signals making them work actively. If it wasn't, then the results were low often even at a higher teacher forces expense.

The right signal for all the students consists of students' most clear understanding that the first, second and nearly every set of the first 25-50% of a course problem are solvable at a given group level. For this reason, we often prefer minimal examples for each particular method (corresponding to an academic hour or two), but more importantly, we have to totally ignore the crude – high-level course structure starting from the easiest – primitive problems, that minimally organized and minimally successful students can solve almost at common sense and limited neighbor & teacher support. So far, all the problems the author has divided into several classes:

1) starting problems (first primitive 4 problems, requiring from 15 minutes up to 1 academic hour for complete solution)

2) normal initial problems (1 academic hour)

3) middle initial problems (up to 2 academic hours)

4) hard but still intuitive problems (visual methods requiring up to 3-4 academic hours)

5) super hard (non-visual, super-hard, and simultaneously low informative problems 2-6 academic hours)

This was not a low-level classification. The main parameters we took into account (except for experimental data) were 1) visuality (more than 70% of people think visually) 2) significance and interest 3) technical difficulty 4) routine work quantity.

Of course, many of these parameters depend on how bright and efficient the teacher 1) presents an algorithm 2) presents its' applied valuability 3) controls the size of the study problem 4) at least what way of calculations and their graphical decoration is used. Still after a minimal experience, we may easily rank all the problems in each subject area by hardness. Then we obtain a table or matrix with such lists, corresponding to one subject in each column.

The difficulty besides opposition visual vs symbolic solution style and solution size primarily depends on the apparatus used. One can arrange the difficulty (see the rank numbers in brackets) by this parameter (this classification is pretty incomplete for the hard level due to combinatorics of possibilities, but mainly due to our primary interest in the easiest–starting methods and the super hard and hard methods alternativelessly should be placed at the end of the learning sequence)

Starting level (0, 0+)

- Scalar inequalities (0)
- Vector inequalities (0+)

Middle level (1, 2)

- Sum operations
- Vector or vector-matrix multiplication (without any equations)
- Sum operations plus inequalities (like in dynamic planning (or programming))
- Primitive finite field (modular) arithmetic
- Elementary quadratic equations
- 2-d linear equations
- Primitive stable point finding iteration methods (like compressive mappings)

Hard level (3, 4, 5)

- General vector equations solution e.t.c.

Super hard level (4-6)

- Non-visual logical or other symbolic operations
- Symbolic operations plus vector equations (at first some simplex-method variants)
- Very complex modular calculations (such as chain ratios in Euclid's algorithm or finite field vector operations like in factor base algorithms etc.)

The independent class is laboratory works having by default twice (or +2) technical difficulty, due to a number of programming operations (and the impossibility to (re-)make the work at home).

The easiest problems are usually based on inequalities, and the next on vector inequalities. These types of problems we observe in 2-4 subject subareas – mainly in game theory (and partially in decision-making) and classical operation research (Prim and Kruskal minimal spanning tree building algorithm). So far, regardless of a faculty or a student group specialization we are in an irresistible way forced to start any general applied mathematics course from the primitive game theory and the most primitive operation research.

IX. Individual Subject Area Structure. How Wide-spread Are Successful Methods in Different Subject Areas

Let's take seldom insertion in the general course of applied mathematics subject - cryptography. In this subject area is a huge number of different algorithms and methods. Still, if we study fast-degree calculation, RSA, Massi-Omura, Diffie-Helman, and Euclid algorithms at low modules like 51 or 33 for RSA, 31 or 97 for Omura, and 997 for Euclid and fast-degree transform we get in less than 8 academic hours (or with factor base method laboratory work (using Excell at some step) 4 more hours) a first substantial acquaintance with the area (all the non-laboratory solutions should be got manually except for manual calculator elementary multiplications).

TABLE I. CRIPTOGRAFY EFFICIENT PROBLEMS LIST.

	Cryptography	price	complexity class
	Small finite field exploration and multiplication table building	1,5	1-
	RSA mod33	1	1-2
	Massy-Omura	2	2
	Fast power exponent over a finite field mod 997	1,5	1,5
	Factor-base method	4	4
	Euclid algorithm, 4 examples	4	4

Next, let's take decision-making support. There is pretty enough use of a linear method of hierarchical analysis (in 2-3 variants), 10-12 criteria calculation, and one purely discrete verbal alternative analysis method (like ZAPROS). Of course, the author is acquainted with ELECTRE-MAUT, etc. methods, but they should be omitted for a set of reasons. So, we have 5-10 academic hours per this subject area.

TABLE II. DECISION-MAKING EFFICIENT PROBLEMS LIST.

	Decision making	price	complexity class
	8-12 Classical criteria	1,5	1-2
	ZAPROS method of verbal analysis	2	2
	hierarchical analysis oversimplified	1,5	1,5
	full hierarchical analysis with a mistake calculation	3,5	2
	hierarchical analysis (fuzzy logic)	1,5	2-

The mentioned examples are enough to understand that rather developed areas may be presented as rather short insertions to a general universal course of applied mathematics. If we switch to widely intersected areas of algorithmizing, narrow operation research, and rest discrete math (for non-mathematicians) we'll see that the number of really conceptual core methods worth course mentioning not exceeds 20 and closer to ten. Approximately the same situation for conceptual classical game-theoretical problems. The exact set, of course, depends on individual preferences, but for classical operation, research one should take the Kruskal (Prim) spanning-tree algorithm (at the right approach it would be the best starting point of the whole applied-mathematics course), minimal route at a simple one-directional graph, graphical linear-programming method, mass-service example, traveling salesman, storage management (may be in the macroeconomic context of cash demand) and project management (13-15 academic hours). One may optionally add simplex methods at least for transport problem (4 hours) and some other applications but this might be too non-successful for a cost–effect ratio. This operation research area is mature so this list is nearly undiscussable.

TABLE III. CLASSICAL OPERATION RESEARCH EFFICIENT PROBLEMS LIST.

	Operation Research	price	complexity class
	Kriuscall algorithm	1	0
	Optimal route at 8-vertex graph	1	1
	Wilson formula	1	1
	Warshall algorithm for a circled weighted chain	1	1
	Wave Dijkstra algorithm for 12-14-vertex graph	2	2
	Net-project management algorithm	3	2
	Some mass service problems	-	2
	Warshall algorithm for a 5-vertice random graph	1	1
	Graphical method for a linear optimization problem	2	2,5
	Traveling salesman problem for 5-vertex graph	4	3-4
	Dual net-project management algorithm	4	3-4
	Ford algorithm	2	2-3
	Transport problem	4	4

For a much less mature game theory subject area we have a standard core of 10-14 classical themes– a game tree without complex information structure, an antagonistic game in dominated strategies, (portfolio) mixed-strategy 2xN game, an antagonistic game based on bi-matrix threats, 3-4 gamers coalitions Shapley solution, (asymmetric) Cournot duopoly, median mechanisms, consequential prisoner-dilemma (10-14 hours), maximally primitive stochastic example, some populational games like the hawk-dove game, optionally job matching (marriage) game, etc.

TABLE IV. CAME THEORY EFFICIENT PROBLEMS LIST.

	Game theory	price	complexity class
	Elementary game-tree	1	0
	Dominated matrix game	1	0
	Hawk-Dove population game	1,5	1,5
	3-4 player Shapley vector for coalition game	1-2	1,5(1+)
	2x2 Stochastic game	1	1
	2xN investor game	2	2
	2x2 Bi-matrix threats	2	2
	(Asymmetric) Cournot duopoly	2	2
	Consequential Prisoners' Dilemma	2	2,5
	Crossroad game	3	3

For discrete math addition, the author used 5-7 basic problems on logic and logical schemes (12 hours).

TABLE V. DISCRETE MATH ADDITION EFFICIENT PROBLEMS LIST.

	Discrete math add	price	complexity class
	Electronical scheme simplification	4	1,5
	3-set intersection problems	1,5	2
	DNF, KNF, PNF for 4 random 2-argument functions and basis formation using Post class identification	2	2
	DNF, KNF, PNF for 4 random 3-argument functions, basis building Post class identification	4	3
	Logical formulas simplification	2	2
	Some other problems		

The rest methods were regarded at the most discussive algorithmizing area. Among them, one should name fast calculations: fast sort, fast multiplication (Karatsuba and optionally other methods), fast matrix multiplication, fast (finite-field) power calculation, and some theoretical problems (at least 17 hours).

TABLE VI. THE ALGORITHMIC ADDS EFFICIENT PROBLEMS LIST.

	Algorithmic add	price	complexity class
	Merge sort for 16 (32) arrays of integer numbers (as a 2-branch recursion example)	1,5	0+
	Karatsuba algorithm	1,5	1,5
	Shtrassen fast matrix multiplication method (as a 7-branch recursion example)	3	2
	Chromatic polynomial for 5-vertex graphs	5	4+
	Advanced numbers multiplication methods	-	3
	Turing machine to conjunction transform	5	4

The author insists on the obligatory inclusion of two more very distinct problems:

1) Elementary financial mathematics (starting from return rate calculation for an elementary project via a quadratic equation and some more standard calculations) (2-3 hours)

2) Linearization of two-dimensional (with second-order polynomials in the right part) non-linear malty-stable differential equation with identification of each equilibrium type (4 academic hours for a 4-equilibrium differential system).

So, we have a potential program with 5+3+7+10+6+6+2~39(40) almost obligatory purely theoretical methods for at least 75-90 hours. One should add here some Excel & C# laboratory works on genetic programming, neural networks, chaos, non-linear dynamics, self-organization, bifurcations and phase-transitions, and crude system cycles for about 100 academic hours more.

Of course, one can add and omit some parts, but the situation is that one hardly finds something worth mentioning if he omitted all the methods we discuss.

Of course, we haven't taken into account optimal coding and information theory, we ignored complex financial mathematics, randomization (though cryptography is about it), we ignored optimization, functional calculation, and ordinary and partial differential equations. Still, nearly all these topics appear generally as sub-methods in the briefly mentioned laboratory works.

X. From Successful Methods to a Successful Course

Next step we should create a whole course program, choosing what material should be taught earlier. Since we are primarily interested in a good start of the whole course, we should arrange the subject columns according to the easiness of their first 2-4 easiest problems (or the first 20% of problems).

In the tab.VII there listed the comparative qualitative difficulty of the first problems at [1] (some easier problems appear after more hard ones for practical purposes due to different interest-complexity ratios)

Thus, in constructing the course sequence, we need to follow from the up-left corner to the down-right in the logic of 'visiting' the easiest and most interesting tasks first. Maybe it's not logical to switch from first scalar inequality algorithms like Kruskal and primitive game-tree convolution to the merge sort (either scalar inequalities) or to maybe some partial ordering decision-making methods (verbal alternative analysis), but at least three or four switches and returns in the marginal left columns are inevitable. As the whole problem set for operation research (OR) and Game column is up to 14-16 problems, including rather had ones, so at each switching, we will solve about 3-5 problems of the current complexity class. The switching frequency is usually maximal at the course beginning.

TABLE VII. First easiest problems arrangement by sub-areas.

#	OR	Game	DM	Alg	Crypt	DataS	DMA	NLS
1	0-	0	1,5	0++	1+	1,5L	2-4	2L
2	1	0+	2	1,5	2	2L	2	2L
3	1	1	1++	3	1,5	3L	2	3L
4	1	1	2	3	2	3L	4	3L
5	2	1	4		4	3L		

#	Column legends	
1	OR	Operation Research
2	Game	Game theory
3	DM	Decision making
4	Alg	Algorithmizing (addition) and theory of algorithms
5	Crypt	Basic cryptography introduction
6	Data	Introduction to data science labs
7	DMA	Discrete math addition
…	NLS	Modeling and non-linear science labs set

The structure of this table is approximately the following

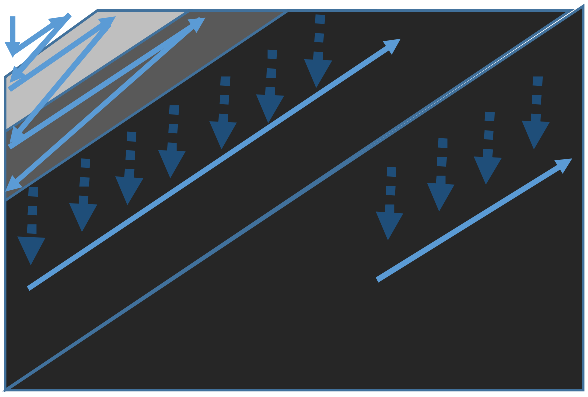

Fig. 2 The complexity-sort-based merging algorithm for general non-engineering (mainly discrete) applied mathematics course sequence construction. By darker we denote more complex tasks (from inequalities in the up-left corner to complex non-graphical and logical algorithms in the opposite direction).

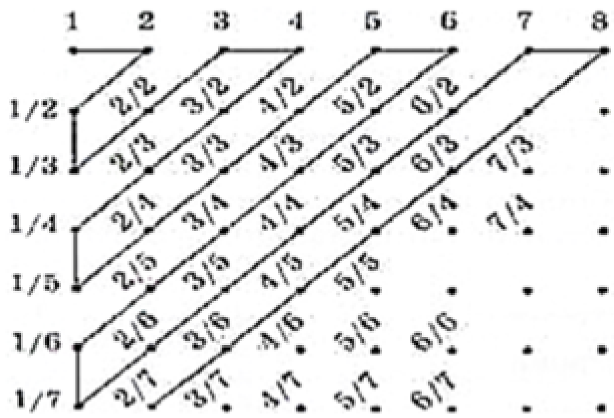

Fig. 3 The famous rational numbers enumeration scheme (used to prove their countability) at external view is quite similar to our complexity-based problem sort approach.

The blue arrow in Fig.2 denotes the most preferable sequence of course study. This may partially remember the merge sort, but most of all it is similar to the classical rational number enumeration procedure when the numbers are classified by the numerator and denominator sum like the following Fig.3

In our case, we have something similar to this scheme but the tasks are sorted by the sum of sub-area rank and internal task-complexity rank (it has got in the given sub-area), maybe with some non-equal weights – in our program the sub-area rank may be considered, for example, twice bigger.

XI. The Applied Mathematics Course As A Version Of General Mathematical Course For Non-Specialists

Due to the 'negative successes' in our education system development, a teacher sometimes gets a problem of explanation of linear algebra at 12 academic hours or 3 2-academic pair meetings. This is the classical case when problem-solving is the best possibility to transmit some knowledge in a limited time. The same problems appear when a teacher should teach all the mathematics in one course. The traditional approach is (roughly speaking) to cut a fundamental university course at some point according to the program's time limit. One of the other possible ways is to extend the applied mathematics course with several problems concerning eigenvalues, linear equations, and matrix operations (and maybe some statistical, etc addition if required). Such methods like linearization of 2-dimensional ordinary differential equation with simple polynomials in the right part with enough number of equilibria, complete canonical hierarchical analysis with a mistake calculation (using eigenvalues), and operation research (game theory) problems requiring linear equation systems solutions may be a not enough but forced by an unprecedented timing constraint solution.

XII. An Obvious Possibilities To Integrate The Applied Mathematics To The General School Programm

Before applied mathematics will become a standard discipline, it should become successful and known as a part of other existing disciplines. There are at least two obvious candidates to implement and legalize corresponding material in the program: school informatics (developing it to some computer science analog) and the concepts of contemporary natural sciences (the Russian for KSE), actively introduced in the Russian educational system. In the first case, it can mean a more fundamental computer science course. In the latter case, this means the mathematization of the KSE discipline.

XIII. Contemporary Tecnology Possibilities

First of all, contemporary technologies make it possible to visualize some aspects of explanation. As the author claims it is usually mainly an evil and only at some proper conditions it's mainly a possibility. This conditions example is described in [2], where we introduce a fast long method learning system based on the integration of malty-parametrical problems, brigade learning, and integrated seminar (simultaneous lection, seminar, and problem-solving) technology. Only if we follow this or not less effective technology of active learning, when each student needs to execute the algorithm with his pencil, we have a moral rule to pose a problem of maximal visualization as a part of such integrated seminar active learning. And especially in this case, there appear very good possibilities for content producers. It is due to the lack of good methods in applied mathematics as a discipline: one should explain much fewer single problems than in fundamental disciplines. Moreover, with the parametrical problem set like [1], one can build an explanation adopted for one-example solution and often (even) in terms of this solution.

XIV. Conclusion

Despite mainly very high method length and complexity granting high difficulty of applied mathematics course running and despite the appropriate study problems shortage we have 40 methods and corresponding problems and 40 labs (that are a topic of a quite distinct discussion) we potentially can teach students for an adequate time the university administration can afford. Besides the proper methods based on integrated seminar methodology [2] (when we integrate lection, seminar, and problem solution), it requires overcoming several myths about course constraints.

We should forget at least for the next several decades of specialized courses for different specializations. More than 2/3 of a material will be common and it is due to the deficit of effective ideas.

We should also definitely reject any ideas of library catalog logic in course time-table planning, but rather we should follow the principle usually at the longer time interval realized at the education as a whole when we from school to the university permanently study one and the same subjects but at deeper (or applying to applied math case - more complex) level.

The other conclusion is that by applying active learning methodic (when method listening and application are integrated into one activity) one can effectively use the presentation media and other technical possibilities. The low spectrum of problems makes it possible to concentrate resources on the corresponding malty-media content creation.

References

[1] Krivosheev O.I. 100 main problems of the applied mathematics [online] Available: https://disk.yandex.ru/d/p6EIeDFvNMSoxg (Russian).

[2] O. I. Krivosheev, F. F. Pashchenko, Y. I. Kudinov, and E. S. Duvanov, "Quasi-Cryptographic Methods in the Management of Educational Tasks," *2022 2nd International Conference on Technology Enhanced Learning in Higher Education (TELE)*, Lipetsk, Russian Federation, 2022, pp. 225-229, doi: 10.1109/TELE55498.2022.9801047.

[3] S.Howison Practical Applied Mathematics, OCIAM Mathematical Institute Oxford University, 2004, 299 pp. [online] Available: http://people.maths.ox.ac.uk/fowler/courses/tech/sdh.pdf

[4] Barrows, H.S.; Tamblyn, R.M. Problem-Based Learning; Springer: New York, NY, USA, 1980.

[5] Rézio, S.; Andrade, M.P.; Teodoro, M.F. Problem-Based Learning and Applied Mathematics. Mathematics 2022, 10, 2862. [online] Available: https://doi.org/10.3390/ math10162862

[6] [online] Available: http://research.microsoft.com/en-us/projects/problemgeneration/default.aspx.

[7] [online] Available: http://research.microsoft.com/en-us/um/people/sumitg/pubs/aaai12-full.pdf.

[8] N.A. Kleshcheva, I.M.Tarasova Projecting of fundamental-applied content of students mathematical education - Fundamental Research Journal) – 2012. – # 3-3. – pp. 586-590;

[9] [online] Available: https://fundamental-research.ru/ru/article/view?id=29761 (Russian).

[10] F.Z.Kaynarov Innovative methods of applied mathematics teaching. [online] Available: https://cyberleninka.ru/article/n/innovatsionnye-metody-prepodavaniya-prikladnoy-matematiki (Russian)

[11] O.E.Akimov Дискретная математика: логика, группы, графы (Discrete Mathematics: Logic, Groups, Graths) [online] Available: http://klassikaknigi.info/akimov-o-e-diskretnaya-matematika-logika-gruppy-grafy/