Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 53-2 (2020) 17326–17331

Automatic control: the natural approach for a quantitative-based personalized education

Steffi Knorn * and **Damiano Varagnolo** **

* *Otto-van-Guericke University Magdeburg, Germany*
** *Norwegian University of Science and Technology, Trondheim, Norway*

Abstract: This paper proposes an engineering-oriented framework that casts the problem of learning as an automatic control problem, and that can ultimately be used to design education activities that autonomously adapt to individual students' abilities, prerequisites, learning goals and other restrictions. The framework leverages on quantitative descriptions of knowledge flows within university programs in terms of Knowledge Components Matrices (KCMs) and Knowledge Flow Graphs (KFGs), that serve as the basis for developing the aforementioned automated approach to personalized education. Essentially, the manuscript proposes to: *1)* combine these descriptions with results from exams and assessments to statistically estimate the learning status of a student; *2)* combine these descriptions with data-driven approaches to derive models of how knowledge ladders logically and in time; *3)* use these two ingredients to automatically design suitable and personalized study activities for a student, given his/her current knowledge status and desired learning outcome. We describe all steps (modelling of the knowledge flows, estimating the current learning status, and derivation of suitable learning activities to close the loop) with formal and control-oriented notation. The paper serves thus the purpose of showing how methods from the field of system theory and control engineering are naturally useful for the implementation of quantitative-based personalized education.

Keywords: university program design, learning model, course design, individualized and personalized education

1. INTRODUCTION

Teaching and learning has been historically often performed by grouping students in classes and subjects, with the underlying assumption that students within a group have the same or similar prior knowledge and learning goals. This approach lowers the workload for teachers, but might be less suitable for individual learners.

The concepts of individualizing, differentiating, and personalizing learning and teaching aims to tailor the learning conditions for each student. More precisely, following the notation suggested in Kerr [2016],

individualization: letting different learners with the same learning goals progress through the same material at different speed;

differentiation: letting different learners with the same learning goals use different materials, methods, instructions, etc., depending on their preferences;

personalization: creating different learning goals for each learner, and making also the progression speed, material and teaching & learning activities different.

Intuitively, if we can adapt education to the unique needs of each learner, this will improve the efficiency of the studying efforts. To enable this without increasing teachers' burdens, digital tools will be needed. From an automatic-control point of view, the most interesting ones come from the two overlapping communities of *educational data mining* and *learning analytics*. More precisely,

educational data mining: developing and applying data mining and statistical methods to build *learner models* (i.e., mathematical models to estimate the current knowledge state of a learner and to predict future performance) starting from information collected within educational settings (see Baker [2014]);

learning analytics: measuring and processing information (collected within educational settings) to model and optimize the learning processes and environments (see Gulbahar and Yildirim [2019]).

These intertwined communities share with the automatic control one the concept of quantitatively modelling the quantities of interest, and deciding how to influence the system, i.e., implement feedback, based on the models. From the control systems perspective, potentially the most interesting usage of these models is implementing *adaptive learning / adaptive teaching* strategies, Turong [2016], Kerr [2016], through Intelligent Tutoring Systems (ITS), Kulik and Fletcher [2016], i.e., digital tools which use algorithms to orchestrate the interactions between the learner and the digital support, and hence deliver customized resources and learning activities for each learner, Mavroudi et al. [2018].

⋆ The research leading to these results has received funding from the European Community through the Erasmus+ project 2019-1-NO01-KA203-060257 "Face It". Corresponding author: S. Knorn, steffi.knorn@ovgu.de

10.1016/j.ifacol.2020.12.1819

The development of these technological tools and aids for teaching have facilitated individualized and personalized teaching specially in technical areas, see Vogel and Klassen [2008], Yan et al. [2016, 2017], and it could be shown that students using the tools achieved better results than students receiving education in a traditional learning environment, see Bahçeci and Gürol [2016].

Summarizing, effectively personalizing learning builds on three main ingredients: (1) a model or an understanding of what has to be learned and its structure, (2) a method to estimate the current learning status of a student and (3) a possibility to adapt the teaching and learning environment and study material for students according to their status.

As for (1), there exists an extensive literature. In Yan et al. [2016, 2017], teachers divided the material for a `C` programming course into smaller parts and concepts. For university courses in general, such detailed information is often not available but exists in form of a more or less broad list of Intended Learning Outcome (ILO) or course / program goals.

In order to estimate the current learning status of a student, i.e., (2) above, in relation to the collection of material to be learned, different tools such as quizzes, tests, exams, questionnaires or even interviews can be used. In order to estimate the learning status in between assessments, the effect of forgetting may be modelled by the Ebbinghaus forgetting curve or suitable extensions, see Lee [2004], or information about which material students were exposed to may be used, see Yan et al. [2016, 2017].

The crucial step after describing which content a student should learn and their knowledge learning status is to adapt the material and learning contents to the student's needs, i.e., (3) above. For instance, the system developed in Yan et al. [2017] gives recommendation to students on which material to study or exercises to solve based on their current learning status. In a slightly different approach, Pavlik and Anderson [2008] suggests a method to schedule practice in order to maximise learning and retention.

There exists many more examples for ITS, but even if ITS are successfully used in some areas, there exist some general shortcomings that need to be addressed. First, the vast majority of ITS in higher education are implemented for specific courses, usually in technical areas, and predominantly for computer science. Hence, suitable solutions for other disciplines need to be found or the existing concepts might need adaptation. As a second aspect, ITS are considered (commercial) tools and in a way separate and with very little similarities to traditional courses at universities. To the best of our knowledge, there exists no universally good approach to bridge the gap and to guide the gradual transformation of existing, traditional courses and programs towards more individualized or personalized learning through ITS. Lastly, the learner models are often rather simple and do not include "dynamics" in the control theory sense.

Summarizing, our opinion is that automatic control may help improving the current ITSs ideation and creation paradigms to facilitate personalized learning.

Summary of contributions We aim to frame how to create and enable personalized learning for entire university programs, building on existing courses and for all areas and disciplines, using a control-engineers-friendly lexicon. We describe a method that allows in principle to automate the process of providing personalized education and to implement "feedback" in the educational process which is closely related to the concept of feedback in automatic control. Importantly, we expect the proposed method to be useful for many if not all university programs.

In other words, the manuscript proposes to tackle the problem of personalizing education using automatic-control and data driven methods. *The authors' intuition is that any attempt to successfully personalize the learning process has to follow a similar approach.* More precisely, personalization always requires to estimate its peculiarities and to tailor the activities to the individual needs. Importantly, this manuscript is a perspective paper: it motivates why and how implementing such an automatic-control-type strategy could radically change education, but does not provide quantitative evidence. The manuscript thus does not contain quantitative claims, but rather the authors' view on how to implement personalized learning in structured education, and why the automatic control community *has* to participate in the development of ITSs.

In summary, this manuscript:

- defines a framework for representing structured programs in a quantitative fashion;
- describes how to use these quantitative representations to estimate the learning status of individual students and to suggest them personalized study activities while updating the learning model in a mathematically formal way; and
- lists a series of mathematical tools / frameworks that can serve the purposes above, moving thus from intuitions to formal tools.

Note that the same system is also suitable to provide higher level feedback to teachers and program boards. Further, the manuscript emphasizes that the strive should be towards developing *data driven* methods, which aim to be less influenced by subjective opinions, traditions and established customs. Indeed, it is anticipated that using numerical evidence and data in order to ultimately steer education and learning processes will contribute to provide learning and teaching experiences that are less affected by conscious or unconscious bias and prejudice about students in specific areas or with specific background.

Organization of the manuscript After summarizing the workflow in Section 2, Section 3 proposes a strategy for representing structured programs in a quantitative fashion. Section 4 discusses how these representations may be used to achieve personalized learning from intuitive perspectives. Sections 5, 6, and 7 then discuss how to achieve this from mathematically formal perspectives. Section 8 finally draws some conclusions.

2. PROPOSED WORKFLOW

The envisioned strategy towards automatic, personalised education is:

Step A) Given a university program, there is a typically implicitly known reference knowledge that students should achieve in time and a minimal learning trajectory that each student should follow. As a first step, this minimal reference knowledge should be opportunely translated into quantitative indications.

Step B) Each individual student is at any time characterized by their knowledge levels, which can be approximated as a quantitative "state", i.e., numerical indications of what a student knows and how well. Hence, from the assessable test results, an estimate of her/his current individual state should be achieved. Collecting performance data over time, moreover will enable to populate a (potentially individual) numerical model of the learning dynamics.

Step C) An opportune IT-based suggestion system can then be implemented that, given the individual estimated status of the student's knowledge and the model of the student's learning dynamics, computes and suggests personalized suitable learning activities that are expected to optimally improve her/his knowledge status.

3. KNOWLEDGE COMPONENTS MATRICES AND GRAPHS

A university program is usually structured and designed with the aim of ensuring that students acquire a desired set of Knowledge Components (KCs) relevant to the program. Consider that the required knowledge for a given university course can be described by a list of KCs (e.g., facts, concepts and procedures) and on which level they need to be mastered / developed (e.g., remember, understand, apply, etc.) when passing the course. One may recognize two types of pieces of knowledge involved in every course: information which should be learned in the course, and information which is a required prerequisite of the course.

Since different aspects build on each other, these KCs are interrelated. We propose to describe these relations within every single course by a so called Knowledge Components Matrices (KCMs), a matrix as in Figure 1, consisting of one row per developed KC, one column per each required and developed KC, and its generic (j, k)-th element being the taxonomic relation between developed KCs j and the required or (different) developed k. Setting element (j, k) to a level on a predefined scale then allows to describe how relevant k is to reach j.

The most practical way of collecting data to build a KCM is, according to the authors' experience, to ask teachers in the program to provide data on their courses. Hence, the KCM is an educated, yet subjective, guess on which causality relations hold among the different pieces of knowledge within a course, possibly averaged by asking several teachers. Importantly, such a KCM can immediately be conceived as a directed, weighted graph, called Knowledge Flow Graph (KFG), representing each KC in a course as a vertex, and every element (j, k) of the related KCM as an edge from vertex j to vertex k with the edge weight being the value of the matrix element. As an example, Figure 2 is the KFG translation of the KCM of Figure 1.

Moreover, since courses in a program tend to require KCs from previous courses, KCMs and KFGs from related courses may be combined, leading to larger matrices/graphs. *Merging all KCMs of a program then translates into a Program-wide Knowledge Components Matrix (PKCM) / Program-wide Knowledge Flow Graph (PKFG) describing the intended knowledge flow within a program.*

		required KCs		developed KCs	
		complex numbers	Riemann integrals	Fourier transforms	Laplace transforms
developed KCs	Fourier transforms	1	2	0	0
	Laplace transforms	2	2	1	0

Fig. 1. Toy example of a hypothetical KCM: its rows list the developed KCs, its columns list both required and developed KCs. Each element (j, k) of this matrix describes how relevant KC k is to reach j. Note that repeating the developed KCs both in the rows and in the columns enables to capture dependencies between KCs developed in the same course.

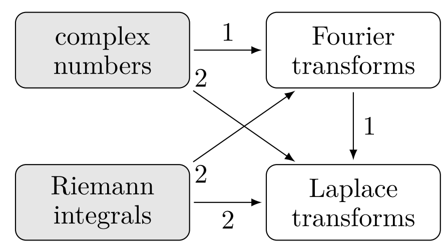

Fig. 2. The KFG corresponding to the KCM shown in Figure 1.

4. PKFGs AS ENABLERS OF AUTOMATIC CONTROL-ORIENTED APPROACHES TO EDUCATION

A comprehensive PKCM and PKFG may be used as an effective tool to enable the introduction of quantitative and data-driven strategies for personalized education of students in the program.

As a first step, in order to describe the current knowledge of a student, one can mathematically interpret a generic PKFG as a random field $\mathcal{K} : (s, n, t) \mapsto \ell$ describing the knowledge level ℓ for the generic student s at time t relative to the piece of knowledge n. The knowledge levels can again be related to a well known taxonomy such as Bloom's or SOLO, see Anderson et al. [2001], Biggs and Tang [2011] and can be multivariate – even if for the purposes of this paper we assume a scalar taxonomy.

Hence, the PKFG can be seen as the field capturing how knowledge levels change in time for the various students. For example, let the vertex $n = 7$ be the concept "complex numbers", and the time be $t = 1$ (i.e., first week at the university). Then if the student s = Ann has a knowledge level 2 for that concept at that time, it means that $K_{s=\text{Ann}}(n = 7, t = 1) = 2$. After 10 weeks of studying math, it may be such that $K_{s=\text{Ann}}(n = 7, t = 10) = 4$. After stopping studying she may also forget things over time, so that $K_{s=\text{Ann}}(n = 7, t = 20) = 3$. In general, we see that $K_s(n, t)$ is a description of how the knowledge for the generic student s changes in time t for the various KCs n within a program.

In this context, exams and tests can be interpreted as operations of sampling from the field $\mathcal{K}$: when students take an exam, they are (noisily) disclosing their knowledge levels relative to some parts of the field. In a sense, every test works as a "sensor" allowing to measure information on some states of the system. For this, exam and test questions must be connected to $\mathcal{K}$.

Further, study activities can be interpreted as an input to the system, as they aim to change the state of at least some parts of the system. However, similar to the exam questions mentioned above, it must be known which KCs are trained or conveyed through a certain study activity and/or which connections between different KCs are used. In practice, students might learn a blend of different KCs from a study activity, even if it might be targeted towards a specific aspect. Hence, study activities will be inputs to the system that include potentially unmeasurable disturbances.

Considering study activities as inputs and exam and test results as measurements of the system serves as a conceptualisation which leads to vision the following automatic-control-like strategy for educating students:

(1) for each student, a PKFG is used as the representation of her/his learning status / state;
(2) every time the student performs an assessable test, the results are used to refine the estimate of the current knowledge status of that individual student;
(3) these results can also be used to improve the accuracy of the learning flows model by considering historic data of the student and/or data from other students in similar situations. This may allow to forecast how the student's state will change due to given learning or study activity;
(4) regularly in time (e.g., daily, weekly, etc.) and based on the estimated state and learning model, an automated system may suggest tailored learning activities to that student, that are intended to be the "best" in the sense suggested by the PKFG.

This framework would thus correspond to a personalized and automatic suggestion system that would complement the current teaching strategy, and would not be limited to single courses or study subjects on technical contents such as programming, as in Yan et al. [2016, 2017].

The analogies and relations between the main terms and concepts from education and automatic control may thus be summarized through the following table:

concepts related to education	*concepts related to automatic control*
student	plant
PKFG	plant model
exam (question)	sensor
exam result	measurement
learning activity	control input

The following sections will then move from discussing the framework intuitively to discussing the three main problems associated to it in a mathematically formal way:

- how to estimate the current knowledge state;
- how to model the learning process; and
- how to suggest personalized learning activities.

5. ESTIMATING THE CURRENT KNOWLEDGE STATUS

There exist different possibilities to estimate the current learning status $K_s(n,t)$ for a given student s, such as final exams as well as concept inventory tests or smaller quizzes. Assume then that an exam or test is designed to assess a set of KCs N_{test}, where a KC in N_{test} is a vertex in the KFG (and thus a scalar component of K_s). To adapt the tests so to be useful to formally estimate $K_s(n,t)$, a meaningful strategy is to tag each of its questions with *(i)* which set of vertices $n \in N_{\text{test}}$ they relate to, and (ii) which level of taxonomy the correct solution is associated to. For this, a strategy is to assume that for all $n \in N_{\text{test}}$ there exists an ℓ_n such that $K_s(n,t) = \ell_n$ with high probability in case at time t student s showed in his/her answer to the question the desired level of knowledge for the vertex n. Further, common wrong answers may be related to a lower taxonomy level of the same set of vertices or allow to estimate the knowledge level of another vertex.

Importantly, information on a KC n may bring information also on KC n' that are known to be related to n. For instance, consider a question about deriving the Fourier transform of a given signal. In the question is correctly solved, the knowledge level of that student relative to the vertex relating to "solving Fourier transform" may be set to a desired value. But, since the use of integral calculus and complex numbers is also required for solving the Fourier integral, the status of that student's vertices relating to these other KCs can also be adjusted, for example by increasing their level from "understanding" to "applying". In case the student cannot solve the question correctly, different possibilities exist on why. For instance, if a student does not know what a Fourier transform is, the related level should be set to a low value. In case the student starts his/her derivations but cannot solve the question due to a lack in integral calculus and/or complex number, the knowledge level of these vertices may instead be adjusted.

It is clear that a single exam question can only cover or examine a subset of the required knowledge. Hence, even all questions in an exam or test usually cannot be used to estimate the current learning status for all nodes n in the field K_s, but only a subset. However, taking into account which prerequisites are implicitly or explicitly used in solving the question may increase the set of nodes for which it is possible to estimate the current learning levels. Also, exams happen at different and discrete time instances t. Hence, various estimates in the graph will have different time stamps and estimates with older time stamps should be accounted with statistically sound strategies, since the learning status might have changed in the mean time. Assessing the knowledge of students often and in smaller time intervals with small tests is hence expected to allow for better measurement of their learning status compared to single, final exams.

Further, students might be able to partly self assess their abilities and report values of their perceived knowledge level. This could be combined by allowing students to request to demonstrate their knowledge to a teacher or TA when they assume to have reached a certain level, as reported in Wrigstad and Castegren [2017].

6. MODELLING THE LEARNING PROCESS

Letting PKFGs be compiled by teachers implies deriving the causality relations among the learning flows solely from the subjective intuitions from the teachers themselves. This strategy is myopic, and should instead eventually be complemented with a data-driven approach towards better modelling how students learn and, therefore, better forecasting how the knowledge status of a student will evolve when subject to learning stimuli.

To estimate the topologies of PKFGs the following information might be used:

- which study or learning activities the student is currently undergoing and which vertices and knowledge levels these are connected to;
- historic data on how the student has progressed (in terms of improvements of knowledge levels in the PKFG) related to his/her study activities in the past;
- historic data on how other students (preferably with similar PKFG) have progressed in terms of improvements of knowledge levels in their PKFG when undergoing similar or the same study activities; and
- known models of learning and retention, Lee [2004].

Formally, the problem is to seek models that in the most general form are

$$K_s(n, t+1) = f\big(K_s(\cdot, t), u(t), n, \{n_{\mathrm{rel}}(n)\}, s, t, \theta\big) \quad (1)$$

where the knowledge level at the next time step $t+1$ (for instance a month, week or day later) depends on the current knowledge levels, the (learning) activities $u(t)$, the KC n, the set of related KCs $\{n_{\mathrm{rel}}(n)\}$, the student s, time t and the model parameters θ.

A slightly more explicit but simplified model capturing the most important phenomena described above could be

$$\begin{aligned} K_s(n, t+1) =& K_s(n,t) + b_{\mathrm{direct}}\big(n, K_s(n,t), \theta_{\mathrm{direct}}\big) u(n,l,t) \\ &+ \sum_{m \in n_{\mathrm{rel}}(n)} b_{\mathrm{rel}}\big(m, n, K_s(n,t), K_s(m,t), \theta_{\mathrm{rel}}\big) u(m,l,t) \\ &- g\big(K_s(n,t), n, \theta_{\mathrm{forg}}, t\big) \end{aligned} \quad (2)$$

where, apart from $K_s(n,t)$, the three terms on the right hand side of (2) describe (i) the positive influence of learning activities directly related to learning n, (ii) the positive effects of learning related KCs and (iii) and negative effects of forgetting learned material over time. The first term after $K_s(n,t)$ contains the learning input, which is a function of item n, the level l and time t, and a factor b_{direct} describing how effective the learning activity is for the knowledge development. Both factors depend on n, since only learning activities contributing to learning n are considered and some aspects might be harder to learn – in these cases b_{direct} might take a low value. Similarly, b_{direct} could decrease for higher achieved learning levels $K_s(n,t)$ since higher taxonomy levels require more effort and are usually harder to learn than lower levels. The parameters θ_{direct} can be obtained from studying data from similar cohorts of students, leading to general parameters and models, or from data of the specific student s, leading to personalized models.

The second term on the right hand side of (2) after $K_s(n,t)$ describes the positive learning effects on $K_s(n,t)$ by study activities directed to related KCs. The factors in the term follow a similar logic as above. However, not only the KC n and its current knowledge level but also the related KC m and the corresponding level are relevant. Note that, it is reasonable to assume that in general b_{rel} will be lower than b_{direct} since it describes indirect learning effects.

In case no direct or related learning activities are undertaken for some time, i.e., $u(n,l,t) = 0$ and $u(m,l,t) = 0$ for all $m \in n_{\mathrm{rel}}(n)$, the knowledge level should intuitively slowly decrease over time, modelling forgetting phenomena. Different models and considerations can be used here. For example, one might assume that achieving a higher learning level leads to forgetting n at a slower rate since a more complex understanding of the knowledge has been achieved, compared to, for instance, only learning something by heart. Also, depending on how important or relevant n might be perceived by s, the rate of forgetting n might change, for example assuming that important and relevant knowledge will be retained longer. Related general or personal model parameters are captured in the parameter θ_{forg}.

It should be noted that the data-driven PKFG modelling problem is expected to be a statistically hard one, with several modelling estimation strategies available, each with specific advantages and disadvantages. In fact, more research is required to understand which model structure is preferable, considering several aspects such as ease of visualisation, complexity, usability, user-friendliness and associated controller design techniques.

7. CLOSING THE LOOP: SUGGESTING PERSONALIZED LEARNING ACTIVITIES

Assume the availability of a knowledge model as a PKFG, estimates of the current learning status of a student, and strategies to predict how knowledge levels will evolve over time as a response to suggested learning activities. All these ingredients can then be combined to *close the loop*, i.e., to develop methods that suggest suitable study activities to achieve some desired objectives. As a matter of fact, different options exists, with three typical approaches discussed below.

• *Minimize the efforts to reach desired knowledge levels within a fixed time window:* a typical situation is that one may want to minimize the effort spent to pass a course. Letting n be a generic KC index, l a learning level index, and $u(n,l,t)$ a certain learning activity, $\phi(n,l,t)$ the effort spent in that specific learning activity $u(n,l,t)$, n_{PLO} the set of relevant KCs, l_n^* the learning levels that should be achieved on these KCs, and T_{learn} the extent of the considered time window. Then the associated optimization problem is essentially

$$\min_u \sum_{t=1}^{T_{\mathrm{learn}}} \sum_{n \in n_{\mathrm{PLO}}} \phi(n,l,t) u(n,l,t) \quad (3)$$

$$\text{such that } K_s(n, t + T_{\mathrm{learn}}) \geq l_n^* \qquad \forall n \in n_{\mathrm{PLO}}. \quad (4)$$

• *Maximize the knowledge increase given a study activity budget,* which aims at maximizing the knowledge increase over all KCs n: this problem involves as a limiting factor the overall sum of all study activities which must not be above an allowed bound. This bound $\overline{U}$ can depend on the student s and be chosen lower to allow students to balance their studies with other duties such as work, taking care of children, voluntary work and other commitments. Further, allowing the weights $\phi(n,l,t)$ to also differ for individual students accounts for individual strengths, weaknesses and preferences. Using the same notation above, this problem may essentially be cast as

$$\max \sum_{n \in n_{\text{PLO}}} \psi(n) \Big(K_s(n, t + T_{\text{learn}}) - K_s(n,t) \Big) \quad (5)$$

$$\text{such that} \sum_{t=1}^{T_{\text{learn}}} \sum_{n \in n_{\text{PLO}}} \phi(n,l,t) u(n,l,t) \leq \overline{U} \quad (6)$$

where a weighting factor $\psi(n)$ can be used to differentiate the importance of the relevant items $n \in n_{\text{PLO}}$.

• *Minimize the time to reach a desired learning goal:* this can be essentially formalised as finding $u(n,l,t)$ that lead in the shortest time to reach a certain predefined set of knowledge levels l_n^*. In practice, only a finite set of study activities will be available: for instance, a limited amount of reading material, consultation time with teachers or TAs or exercise questions. In its essence, however, the problem may be formalized as

$$\min T_{\text{learn}} \quad (7)$$

$$\text{such that } K_s(n, t + T_{\text{learn}}) \geq l_n^* \quad \forall n \in n_{\text{PLO}} \quad (8)$$

$$\text{and} \sum_{t=1}^{T_{\text{learn}}} \sum_{n} \phi(n,l,t) u(n,l,t) \leq \overline{U}. \quad (9)$$

8. CONCLUDING REMARKS

Enabling personalised learning and education at the university level is expected to contribute to more time efficient education and help individual students express their best, thus leading to lower drop-out rates. Over the last decades, several software tools have been developed to handle the large amount of information associated to education, and some efforts are currently being placed to leverage this availability of information for the purpose of individualizing and personalizing education.

It is a widely accepted claim that using data driven methods (that might both improve the learning experience and lead to more objective education systems) requires sound quantitative methods, and that these require in their turn sound mathematical foundations. In other words, data-driven pedagogical efforts have to tightly embrace rigorous statistical modelling.

However, the principal claim of the authors is that statistics is not enough, and that it shall be complemented with control theory: individualization and personalization, indeed, mean adapting to students' needs and goals, and this *necessarily* passes through blending three ingredients: *a)* estimating what is the current knowledge status of the various students (i.e., the initial point), *b)* estimating how their knowledge levels ladder in time (i.e., a model of the learning process), *c)* estimating how different learning activities will (individually) affect their knowledge status (i.e., a forecast of the effects of doing some actions). And these, essentially, are the classical ingredients used in systems theory and automatic control.

REFERENCES

Anderson, L. W., Krathwohl, D. R., Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R., Raths, J., and Wittrock, M. C. (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. *White Plains, NY: Longman.*

Bahçeci, F. and Gürol, M. (2016). The effect of individualized instruction system on the academic achievement scores of students. *Education Research International.*

Baker, R. S. (2014). Educational data mining: An advance for intelligent systems in education. *IEEE Intelligent Systems*, 29(3):78–82.

Biggs, J. and Tang, C. (2011). *Teaching for Quality Learning at University.* Maidenhead, UK: Open University Press.

Gulbahar, Y. and Yildirim, D. (2019). Towards an adaptive learning analytics framework. In *Society for Information Technology & Teacher Education International Conference*, pages 1025–1032.

Kerr, P. (2016). Adaptive learning. *ELT Journal*, 70(1).

Kulik, J. A. and Fletcher, J. (2016). Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of Educational Research*, 86(1):42–78.

Lee, M. D. (2004). A Bayesian analysis of retention functions. *Journal of Mathematical Psychology*, 48(5):319–321.

Mavroudi, A., Giannakos, M., and Krogstie, J. (2018). Supporting adaptive learning pathways through the use of learning analytics: developments, challenges and future opportunities. *Interactive Learning Environments*, 26(2):206–220.

Pavlik, P. I. and Anderson, J. R. (2008). Using a model to compute the optimal schedule of practice. *Journal of Experimental Psychology: Applied*, 14(2):101–117.

Turong, H. M. (2016). Integrating learning styles and adaptive e-learning system: Currentd evelopments, problems and opportunities. *Computer in Human Behavior*, 55:1185–1193.

Vogel, D. and Klassen, J. (2008). Technology-supported learning: status, issues and trends. *Journal of Commuter Assisted Learning.*

Wrigstad, T. and Castegren, E. (2017). Mastery learning-like teaching with achievements. In *SPLASH-E.*

Yan, Y., Hara, K., Kazuma, T., and He, A. (2017). A method for personalized c programming learning contents recommendation to enhance traditional instruction. In *IEEE 31st International Conference on Advanced Information Networking and Applications.*

Yan, Y., Hara, K., Nakano, H., Kazuma, T., and He, A. (2016). A method to describe student learning status for personalized computer programming e-learning environment. In *IEEE 30th International Conference on Advanced Information Networking and Applications.*