

e-mentor

DWUMIESIĘCZNIK SZKOŁY GŁÓWNEJ HANDLOWEJ W WARSZAWIE
WSPÓŁWYDAWCA: FUNDACJA PROMOCJI I AKREDYTACJI KIERUNKÓW EKONOMICZNYCH

2015, nr 3 (60)



E. Olejarz-Mieszaniec, *Strukturalizacja wiedzy w adaptacyjnym procesie komputerowego wspomaganie kontroli dydaktycznej*, „e-mentor” 2015, nr 3(60), s. 25–30, <http://dx.doi.org/10.15219/em60.1183>.

Strukturalizacja wiedzy w adaptacyjnym procesie komputerowego wspomaganie kontroli dydaktycznej



Ewa Olejarz-Mieszaniec

Artykuł prezentuje koncepcję wykorzystania strukturalizacji w procesie adaptacyjnego testowania wiedzy studenta. Testowanie adaptacyjne polega na dopasowywaniu przebiegu testu do umiejętności studenta i dokładnym badaniu tych zagadnień, które opanował w największym bądź najmniejszym stopniu, aby upewnić się, czy udzielona odpowiedź nie była przypadkowa. Strukturalizacja wiedzy w adaptacji procesu testowania służy dobieraniu właściwych proporcji pytań ze zbioru kontrolowanych dziedzin, tak aby znajomość każdej z nich została rzetelnie oceniona, umożliwiając jednocześnie sterowanie procesem zadawania pytań w zależności od przebiegu testu. Zaprezentowane w artykule algorytmy, realizujące oba te założenia, są ciągle testowane i rozwijane przez autorkę, choć już teraz ich wykorzystanie w procesie kontroli nauczania daje bardzo wiarygodne wyniki, nieosiągalne w klasycznym testowaniu komputerowym, stając się alternatywą dla czasochłonnego egzaminowania ustnego.

Podobnie jak w wielu dziedzinach działalności ludzkiej, również w procesie uczenia się i nauczania rola komputerów i technologii informatycznych staje się coraz większa. Obok powszechnego już poszukiwania informacji naukowych w internecie oraz różnych technik nauczania na odległość, komputer coraz częściej wykorzystywany jest do weryfikacji i oceny stopnia opanowania określonych obszarów wiedzy.

Testowanie komputerowe, ze względu na różnorodność metod organizacji wiedzy, nie tylko pozwala na udoskonalanie dziedzinowych banków pytań, lecz również na zróżnicowanie sposobów ich

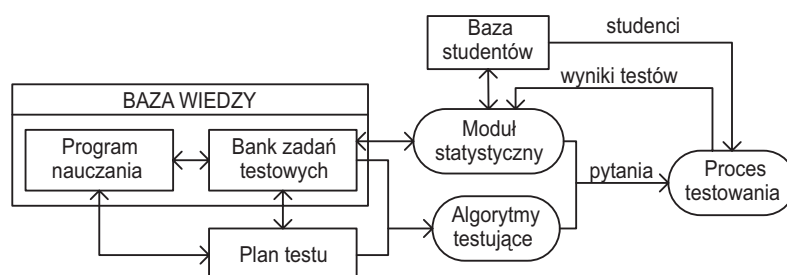
wykorzystania, co stwarza możliwości indywidualizacji procesu kontroli dydaktycznej oraz całego procesu nauczania. Indywidualizacja procesu testowania sprowadza się do wykorzystania adaptacyjnych metod kontroli wiedzy z wykorzystaniem nowoczesnych technologii komputerowych. Pozwalają one na dynamiczne, działające w czasie rzeczywistym dostosowanie kolejnych kroków testu do osiągnięć danej osoby, dzięki czemu testy mogą zostać indywidualnie dopasowane do każdego uczącego się.

Niniejsze opracowanie prezentuje autorski model systemu testującego, składający się z następujących elementów (rys. 1):

- bazy wiedzy obejmującej program nauczania i bank zadań testowych,
- bazy studentów, przechowującej informacje o studentach i rozwiązywanych przez nich testach,
- modułu testującego-uczącego, działającego w oparciu o wewnętrzne algorytmy testujące oraz zdefiniowany przez nauczyciela plan testu,
- modułu statystycznego przetwarzającego zgromadzone wyniki testów, który dzięki możliwości automatycznej weryfikacji parametrów zadań pozwala na bieżąco doskonalić bank zadań testowych.

Autorka skupia się w niniejszym artykule na strukturalizacji bazy wiedzy oraz opracowanych algorytmach, które w zależności od potrzeb nauczyciela pozwalają na dobór najwłaściwszego dla danej sytuacji algorytmu generowania testów.

Rysunek 1. Model systemu testującego



Źródło: opracowanie własne.

Strukturalizacja bazy wiedzy

Strukturalizacja to proces polegający na wyodrębnieniu struktury, czyli na takim ukształtowaniu i uporządkowaniu elementów, aby stworzyły układ hierarchiczny. Również w edukacji mamy do czynienia z procesem strukturalizacji, np. w przypadku programu nauczania czy podręcznika edukacyjnego¹ – można zaobserwować, że główne rozdziały podręcznika składają się z podrozdziałów, a te z kolei z następnych podrozdziałów, przy czym każdy z nich może mieć różną objętość i wagę.

Strukturalizacja programu nauczania proponowana przez autorkę polega na utworzeniu zbioru hierarchicznie uporządkowanych tematów (dziedzin lub kategorii zadań) wyznaczających treść nauczania. Program nauczania jest reprezentowany przez graf składający się ze skończonego, niepustego zbioru wierzchołków oraz zbioru krawędzi łączących wierzchołki, który w praktyce najczęściej przyjmuje postać drzewa. Relacje łączące poszczególne wierzchołki grafu wyznaczają jego krawędzie. Są to relacje zawierania dla poszczególnych dziedzin nauczania. Z uwagi na to, że różne tematy mogą być bardziej lub mniej istotne bądź obszerne, wprowadzone zostały wagi grafu, nazwane parametrami ważności, przyjmujące wartości w przedziale (0,1), których suma w każdym węźle jest równa 1. Przykładowo $w(D11 \rightarrow D1)$ (rys. 1) jest względnym parametrem ważności stanowiącym szacunkową wartość ułamkową D11 w stosunku do D1.

Dziedziny, stanowiące wierzchołki grafu, są powiązane z zadaniami testowymi (umieszczanymi w banku zadań testowych) poprzez parametr pokrycia r , będący stopniem odwzorowania kontrolowanych zagadnień z danej dziedziny w zadaniu testowym. Interpretacja tego parametru jest następująca: jego wartość oznacza prawdopodobieństwo, z jakim można stwierdzić, że student opanował materiał z określonej dziedziny, odpowiadając prawidłowo na zadane pytanie. Jeżeli współczynnik przyjmuje wartość 0,2 – oznacza to, że

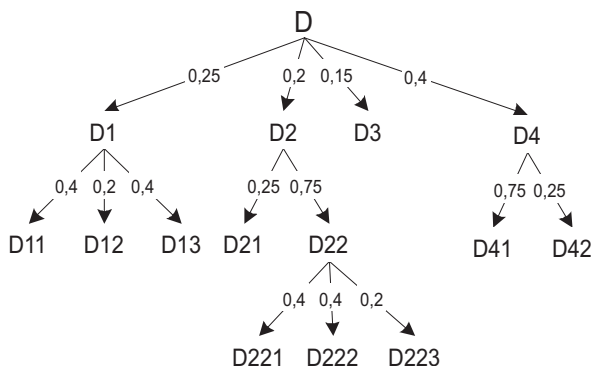
należy wygenerować 5 zadań o takiej samej wartości r , aby mieć stuprocentową pewność, że dana poddziedzina została sprawdzona.

Ważne jest, aby poszczególne zadania były powiązane z dziedzinami na możliwie jak największym poziomie szczegółowości, co umożliwia później precyzyjny wybór testowanego zagadnienia. Na podstawie wartości parametrów r dla zbioru pytań oraz parametru ważności w można obliczyć odpowiednie wartości parametru r dla dziedzin nadrzędnych.

W przypadku gdy zadanie sprawdza wiedzę z kilku poddziedzin i suma wszystkich wartości r określonych dla tego zadania jest różna od 1, obliczana jest tzw. względna wartość parametru pokrycia, spełniająca warunek sumowania się wszystkich wartości parametrów r do 1. Wartość ta jest wykorzystywana w algorytmach tworzenia testów.

Ponadto, oprócz parametru pokrycia, pojedyncze zadanie testowe jest opisane przez współczynnik łatwości zadania testowego, przyjmujący wartość w przedziale (0, 1). Dla zadań najtrudniejszych współczynnik ten przyjmuje wartość bliską 0. Jego wartość początkowo szacuje autor zadania, a następnie, w miarę pojawiania się odpowiedzi, jest ona obliczana i weryfikowana w banku zadań zgodnie z jego definicją i przyjętym wzorem². Współczynnik łatwości zadań pozwala wygenerować wersje równoległe testu, czyli takie, które będą miały zbliżony poziom trudności, ale przede wszystkim odgrywa istotną rolę w module adaptacyjnym działającym w oparciu o model Rascha, mając na celu obliczanie poziomu osiągnięć studenta. Testowanie adaptacyjne jest ściśle związane z koniecznością analizowania wyników i parametrów wygenerowanych już zadań, gdyż na ich podstawie podejmowana jest decyzja o wyborze kolejnych. Aby było to możliwe, konieczne jest przygotowanie odpowiedniego banku zadań testowych oraz strukturalnego programu nauczania. Automataczne doskonalenie banku zadań testowych polega na ciągłej aktualizacji wskaźników łatwości zadania, co pozwala wyeliminować pytania zbyt łatwe bądź za

Rysunek 2. Przykładowy program nauczania z wartościami parametrów ważności (w)



Źródło: opracowanie własne.

¹ W. Okoń, *Nowy słownik pedagogiczny*, Wydawnictwo Akademickie ŻAK, Warszawa 2008, s. 372.

² B. Niemierko, *Pomiar dydaktyczny kształcenia*, WSiP, Warszawa 1999.

trudne, czyli takie, które nie mają żadnego udziału w różnicowaniu oceny studentów.

Przedstawiona w dużym skrócie struktura powiązań programu nauczania z bankiem zadań testowych odgrywa kluczową rolę w tworzeniu planu testu, w przypadku którego warunki początkowe określa nauczyciel, a za dobór odpowiednich zadań testowych odpowiedzialny jest właściwie zdefiniowany algorytm. Możliwości zastosowania gotowych algorytmów oraz stworzenia własnych będą tematem dalszej części artykułu.

Algorytmy testowania wiedzy studenta w oparciu o strukturalizację

Ze względu na sposób generowania pytań testy wykorzystujące technologie komputerowe można podzielić na:

- testy bierne – pytania są zadawane ściśle według określonego przez nauczyciela planu; tworzone są wersje równoległe testów – o jednakowym stopniu trudności; odpowiedzi na poprzednie pytania nie mają wpływu na przebieg testu,
- testy adaptacyjne – rozwiązanie wcześniejszych zadań ma wpływ na zadania następne, które zostają wygenerowane tak, aby możliwie najdokładniej zbadać poziom wiedzy studenta i przetestować określone partie materiału; ten rodzaj testów ma symulować egzaminowanie ustne, w którym np. po błędnej odpowiedzi zadawane jest pytanie pomocnicze.

Idea komputerowego testowania adaptacyjnego³ opiera się na teorii IRT (*Item Response Theory*) – teorii odpowiedzi na zadanie wykorzystującej matematyczne modele probabilistyczne do wyrażania zależności między odpowiedziami na zadania testowe a poziomem wiedzy testowanych. Model Rascha to jednoparametrowy model logitowy określający rozkład prawdopodobieństwa udzielenia przez studenta prawidłowej odpowiedzi na zadane pytanie na podstawie współczynnika łatwości zadania oraz wyznaczanego po każdej udzielonej odpowiedzi parametru poziomu osiągnięć studenta⁴. Model ten doskonale się sprawdza w przypadku testów kontrolujących jedną dziedzinę, natomiast w przypadku testów wielodziedzinowych pojęcie adaptacji powinno zostać ujęte w szerszym aspekcie.

Adaptacja rozszerzona przez autorkę może dotyczyć:

- wyników odpowiedzi studenta – w zależności od jego wcześniejszych odpowiedzi można zmniejszyć lub zwiększyć poziom trudności zadań czy też zadać pytanie pomocnicze z tej samej dziedziny, jeśli student udzielił nieprawidłowej odpowiedzi, aby upewnić się, czy

nie rozumie danego zagadnienia, czy może przypadkowo nie odpowiedział prawidłowo na zadane pytanie;

- przynależności zadań testowych do określonych dziedzin w drzewie nauczania – gdy nauczyciel określi tylko sumaryczną liczbę zadań w teście, system na tej podstawie określa proporcje pytań, jakie mają przypaść na dane poddziedziny, o ile odpowiednia liczba pytań istnieje w banku testów;
- poziomu trudności testu – na podstawie zadanego poziomu trudności testu (zależnego od współczynników łatwości poszczególnych zadań) system dobiera kolejne zadania tak, aby utrzymać zadaną wartość dla wygenerowanego testu.

Istnieją dwie możliwości tworzenia planu testu adaptacyjnego:

- zdefiniowanie własnego algorytmu przebiegu testu na podstawie warunków początkowych oraz zdefiniowanych akcji przebiegu testu,
- określenie warunków początkowych (czas trwania testu, liczba zadań, zakres materiału) oraz wybór jednego z dwóch zaimplementowanych algorytmów tworzenia testu, które zostaną omówione w dalszej części artykułu.

W celu stworzenia własnego algorytmu testu należy zdefiniować poszczególne akcje i uwarunkowania kolejności ich wykonania, które mogą stwarzać nowe możliwości przebiegu testu. Można wyróżnić cztery rodzaje akcji⁵:

- zawierającą przekierowanie do zadania lub zbioru zadań,
- zawierającą przekierowanie do innej akcji,
- warunkową – w zależności od określonych w akcji warunków przekierowuje ona do zadania (pytania) lub akcji,
- akcję zakończenia testu.

Dzięki akcjom warunkowym kolejność wykonywanych akcji można uzależnić np. od wyników rozwiązań zadań z poprzedniej akcji.

Akcja T1: Zadaj pytanie z kategorii D1. Przejdź do akcji TW2.

Akcja warunkowa TW2: Jeżeli akcja została wykonana mniej niż 3 razy, przejdź do akcji T1, w przeciwnym wypadku przejdź do akcji T3.

Taki zapis oznacza, że z kategorii D1 losowane są trzy pytania, za co odpowiedzialny jest warunek zawarty w akcji TW2.

Akcja warunkowa umożliwia uzależnienie przebiegu testu między innymi od następujących warunków:

- liczby zadanych pytań,

³ CAT – *Computerized Adaptive Testing*.

⁴ F.B. Baker, *The Basic of Item Response Theory*, ERIC Clearinghouse on Assessment and Evaluation, New York 2001.

⁵ E. Olejarsz-Mieszaniec, S. Bieniasz, K. Cetnarowicz, *Język definicji testu i jego możliwości w zastosowaniu do e-nauczania*, [w:] A. Wierzbicka (red.), *Akademia on-line. Vol. 2*, Wyższa Szkoła Humanistyczno-Ekonomiczna, Łódź 2006, s. 161.

- czasu, który upłynął od początku testu,
- liczby zdobytych przez studenta punktów.

Poza tym informacja o wyniku bieżącej odpowiedzi na zadane pytanie pozwala przekierować przebieg testu do różnych akcji, które z kolei mogą kierować do trudniejszych albo łatwiejszych zadań, a tym samym indywidualizować proces kontroli nauczania.

Jeśli test ma uwzględniać nie tylko strukturę hierarchiczną kategorii pytań (programu nauczania), lecz także parametry ważności oraz współczynniki łatwości pytań, plan testu można uprościć do określenia warunków początkowych (np. ustalenia tylko limitu pytań oraz nazw dziedzin, które ma obejmować, ewentualnie średniego poziomu trudności testu), natomiast cała odpowiedzialność za jakość wygenerowanych testów przechodzi na algorytm – system sam dobiera proporcję i rodzaj zadawanych pytań.

Bardzo ważne jest dokładne sprecyzowanie ograniczeń kończących test. Mogą nimi być:

- maksymalny czas całkowity testu (*Time*),
- maksymalna liczba wszystkich zadań w teście (*N*),
- wyznaczenie do skontrolowania zadanych partii materiału (zbioru poddziedzin – *Dt*).

Można na przykład zdefiniować test trwający 20 min. o dowolnej liczbie zadań, dobieranej zgodnie z układem programu nauczania (lub jego części), jak również test trwający najwyżej 30 min., składający się z maksimum 20 zadań, obejmujący równomiernie (system generuje proporcjonalnie po tyle samo zadań) dziedziny D1–D5.

Głównym założeniem pierwszego algorytmu jest dobór proporcji pytań z określonych dziedzin w oparciu o strukturę programu nauczania, z którym te pytania są powiązane. Nauczyciel definiuje liczbę pytań w teście (*N*) oraz zbiór poddziedzin (*Dt*), z których każda może mieć różny parametr ważności, a system na podstawie drzewa nauczania wylicza proporcje pytań przypadających na każdą z dziedzin

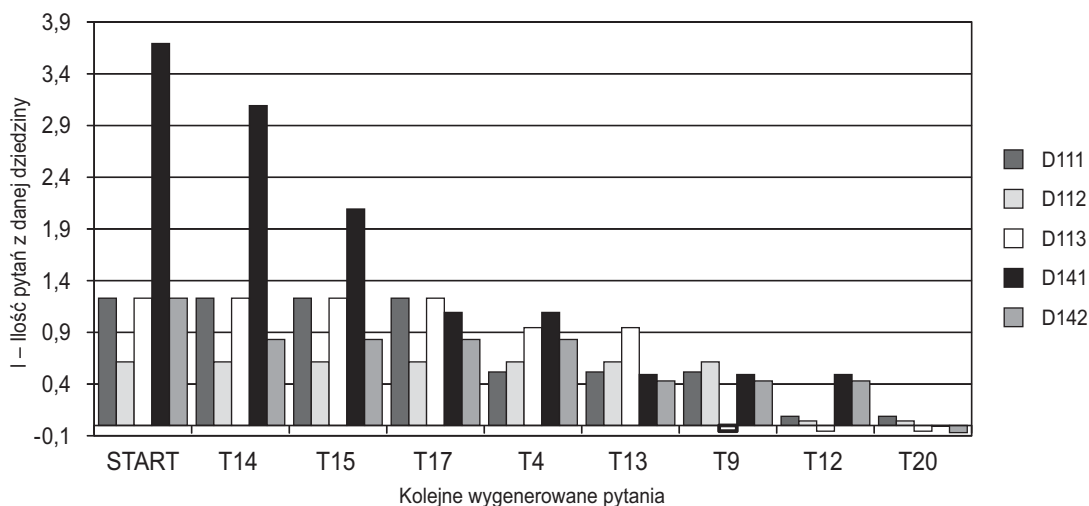
(parametr pokrycia – *l*) oraz generuje zbiór zadań testowych (*Tn*) spełniający określone warunki początkowe. Po każdym zadaniem (wygenerowanym) pytaniu, uwzględniając możliwości przynależenia pytania do kilku poddziedzin, system modyfikuje wartości *l* oraz zbiór *Tn*. Jeżeli drzewo nauczania skonstruowane jest tak, że pewne poddziedziny mają zdecydowanie większe wartości parametrów ważności od pozostałych (np. D141 w stosunku do D112 na rys. 3), proporcje pytań do wygenerowania w teście, czyli wartości parametru *l* dla poszczególnych dziedzin, również będą nierównomiernie rozłożone ($l(D141) = 3,6$ oraz $l(D112) = 0,6$), a w pierwszej kolejności będą zadawane pytania z dziedzin, dla których wartości *l* są największe (w tym przypadku z D141).

Wartości *l* są pomniejszane po każdym zadaniem pytaniu, zgodnie z jego wartościami pokrycia określonych dziedzin. W większości przypadków nie są to liczby całkowite i nie są one zaokrąglane do części całkowitych, stąd też wzięło się określenie ilości bądź proporcji, a nie liczby zadań testowych. Algorytm dąży do sytuacji, w której wartości parametru pokrycia (*l*) dla wszystkich kontrolowanych dziedzin przyjmą wartość bliską 0 w granicach dopuszczalnego błędu (rys. 3).

Czasem może zaistnieć potrzeba przeegzaminowania studenta równomiernie z każdej poddziedziny, niezależnie od ustalonych parametrów ważności (*w*) dla programu nauczania. W tym celu został stworzony alternatywny algorytm, który dobiera kolejne zadania testowe tylko na podstawie wartości pokrycia dziedzin przez poszczególne pytania z banku zadań testowych.

Po każdym zadaniem pytaniu ze zbioru pytań (*Tn*) (wygenerowanego zgodnie z aktualnymi warunkami) względne parametry pokrycia dziedzin (*r*) związanych z tym pytaniem są sumowane. Po osiągnięciu wymaganej sumy (*s*) dla danej dziedziny w zbiorze (*Tn*) nie znajdują się pytania dotyczące skontrolowa-

Rysunek 3. Przykładowy rozkład zmian wartości *l* dla danych dziedzin dla kolejno generowanych pytań w trakcie trwania testu



Źródło: opracowanie własne.

nej dziedziny. Algorytm ten w głównej mierze dąży do równomiernego rozkładu zadawanych pytań z poszczególnych dziedzin, dzięki czemu istnieje możliwość zakończenia testu jeszcze przed osiągnięciem maksymalnej liczby zadawanych pytań czy maksymalnego czasu przeznaczanego na rozwiązanie testu – w momencie, kiedy dla wszystkich zadanych dziedzin (D_t) sumaryczna wartość stopnia pokrycia (s) zmieści się w granicach błędu górnej granicy pokrycia (Z).

O ile w przypadku pierwszego algorytmu proporcje tematyczne zadawanych pytań są zależne od konstrukcji i wyważenia drzewa nauczania, o tyle w przypadku tego algorytmu na zestawy wygenerowanych pytań znacząco wpływają wartości parametrów pokrycia pytań (r) w banku testów określających ich przynależność do dziedzin drzewa nauczania. Niewłaściwy dobór tych parametrów oraz ich niekonsekwentne określanie może przynieść nieoczekiwane rezultaty podczas generowania pytań testowych. Przykładowy przebieg testu z wykorzystaniem drugiego algorytmu i rozkład zmian wartości pokrycia danej dziedziny przedstawia rysunek 4. Zadanie 8 pytań pozwoliło na sprawdzenie wszystkich 5 dziedzin. Warto również zwrócić uwagę, że pytanie T8 sprawdza D142 oraz D112.

W obu zaprezentowanych algorytmach zaprojektowany został moduł adaptacji działający w oparciu o IRT (model Rascha), czyli dopasowania poziomu trudności kolejnych pytań do wyliczanego na bieżąco parametru poziomu osiągnięć studenta. Moduł adaptacji dla wielodziedzinowego testu interaktywnego skupia się na właściwym dla danego algorytmu pokryciu pytaniami zadanych poddziedzin oraz stara się dopasować poziom trudności zadań testowych do umiejętności studenta, generując dodatkowe pytania z poddziedzin, z których udzielił on wcześniej nieprawidłowej odpowiedzi, w celu dokładnego i rzetelnego zweryfikowania jego faktycznej wiedzy. Aby testowanie adaptacyjne było możliwe, w zbiorze pytań możliwych do zadania musi się znaleźć dostateczna liczba pytań o zróżnicowanym wskaźniku łatwości. Moduł adaptacji może zostać włączony w obu algorytmach i bazuje na zbiorze pytań (T_n) generowanym w wyniku działania przedstawionych wcześniej procedur.

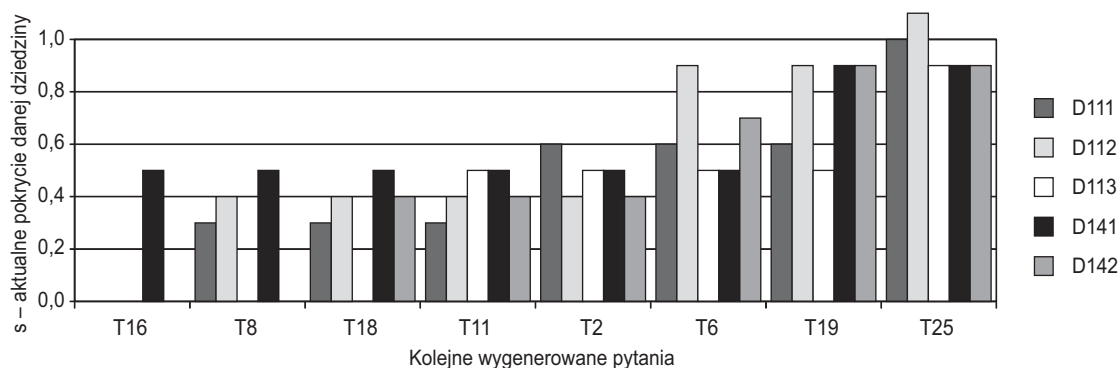
Podsumowanie

Zaprojektowane algorytmy dla systemu testującego, oparte na strukturalizacji wiedzy, stanowią rozwiązanie niestosowane dotychczas w tego typu systemach. Zasadniczym atutem proponowanego rozwiązania jest jego uniwersalność, czyli możliwość otrzymania zarówno wersji równoległych testu o jednakowym poziomie trudności, jak i stworzenia indywidualnych planów testów, których przebieg jest w pełni uzależniony od kolejnych odpowiedzi studenta lub struktury drzewa nauczania. Jest to możliwe dzięki temu, że zastosowana w modelu adaptacja nie ogranicza się tylko do dopasowania poziomu trudności zadań do umiejętności studenta, lecz rozpatrywana jest również w aspekcie dziedzinowym, który pozwala na dobór zadań testowych w zależności od zdefiniowanej strukturalizacji bazy testów.

Przedstawione w artykule algorytmy zostały zaimplementowane i przetestowane w Katedrze Informatyki AGH na zajęciach z programowania w języku LISP. Uzyskane wyniki pozwalały na udoskonalanie parametrów banku zadań testowych, a przetestowane algorytmy skonfrontowane z testami tradycyjnymi i odpowiedziami ustnymi przyniosły zamierzone rezultaty. Podobne wyniki można było uzyskać, zadając mniejszą liczbę precyzyjnie dobranych pytań.

Rozwiązanie takie niesie ze sobą różne problemy, gdyż opracowanie komputerowego testu adaptacyjnego jest pracochłonne. Testy te wymagają korzystania z dużych banków zadań testowych, a po ich przygotowaniu trzeba uzyskać dużą liczbę rozwiązań każdego z nich, w celu określenia poziomu ich trudności. Porównując jednak wady i zalety adaptacyjnego testowania komputerowego, należy przewidywać dalszy rozwój w tej dziedzinie. Z uwagi na możliwości indywidualizacji procesu kontroli nauczania testowanie to ma cechy nieosiągalne dla klasycznego testowania komputerowego, a biorąc pod uwagę ciągły postęp technologii komputerowej, można przypuszczać, że adaptacyjne testowanie komputerowe z biegiem czasu stanie się podstawowym narzędziem pomiaru osiągnięć uczących się.

Rysunek 4. Przykładowy rozkład zmian wartości s aktualnego pokrycia dziedzin dla kolejno generowanych pytań



Źródło: opracowanie własne.

Bibliografia

Baker F.B., *The Basic of Item Response Theory*, ERIC Clearinghouse on Assessment and Evaluation, New York 2001.

Okoń W., *Nowy słownik pedagogiczny*, Wydawnictwo Akademickie ŻAK, Warszawa 2008.

Niemierko B., *Pomiar dydaktyczny kształcenia*, WSiP, Warszawa 1999.

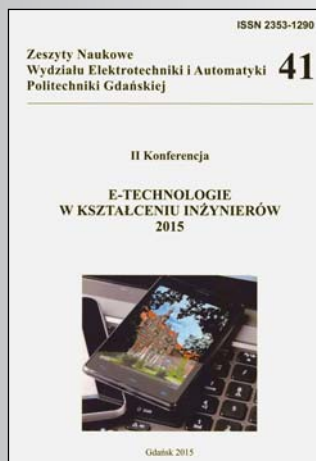
Olejarz-Mieszaniec E., Bieniasz S., Cetnarowicz K., *Język definicji testu i jego możliwości w zastosowaniu do e-nauczania*, [w:] Wierzbicka A. (red.), *Akademia on-line. Vol. 2*, Wyższa Szkoła Humanistyczno-Ekonomiczna, Łódź 2006.

Knowledge structuralization in computerized adaptive testing and assessment of teaching process

Computerized adaptive testing is a form of computer-based test that adapts to the student's ability level. System successively selects questions for the purpose of maximizing the precision of the exam based on what is known about the examinee from previous questions in order to determine whether the answer given was not accidental. In turn, the task of knowledge structuralization is to organize the elements of the educational program to set up a hierarchical structure. This paper proposes a combination of two solutions – the concept of applying structuralization to students knowledge adaptive testing process. The knowledge structuralization in adapting testing process has a purpose of choosing the right proportions of questions from a set of controlled topics, so that each of them was well assessed and, at the same time, taking under consideration the computed level of student achievements. Such task is very difficult, even impossible to achieve with a small and a deficient tasks bank. Using well-defined relationship between the teaching program and question bank, capabilities of the automatic modification of test tasks easiness factor, and thus excluding from the bank questions which do not contribute anything to the process of assessment, and finally using the appropriate selection and configuration of testing algorithm, teacher can achieve very high possibility to create varied adaptive tests. Presented in this paper algorithms are constantly tested and developed by the author, but even now their usage in monitoring teaching process gives very reliable results, unattainable with classic computerized testing, becoming an alternative to time-consuming oral examination.

Autorka jest doktorem nauk technicznych w dyscyplinie informatyka, adiunktem na Akademii Górniczo-Hutniczej w Krakowie. Od wielu lat zajmuje się problematyką e-edukacji. Jej zainteresowania badawcze dotyczą głównie możliwości adaptacji, czyli indywidualizacji procesu nauczania i kontroli studentów.

POLECAMY



E-technologie w kształceniu inżynierów

„Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej” 2015, nr 41

Zapraszamy do zapoznania się z „Zeszytami Naukowymi Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej” zawierającymi zestaw referatów z II Krajowej Konferencji pt. *E-Technologie w kształceniu inżynierów. Otwarcie na nowe wyzwania – wybieramy MOOC?*, która odbyła się 30 kwietnia 2015 roku. Poruszane zagadnienia to m.in. zastosowanie GeoGebra w nauczaniu matematyki, platformy cyfrowe Arduino i Raspberry, doskonalenie jakości kursów e-learningowych, Survey Simulator i Wirtualne Laboratorium.

Publikacja jest dostępna w wersji elektronicznej na stronie: <http://www.ely.pg.gda.pl>.

Jarosław Sosnowski, Łukasz Nowakowski, *Elektroniczne giełdy transportowe* Difin, Warszawa 2015

Celem autorów było przybliżenie zagadnienia elektronicznych giełd transportowych, które pomimo wielu zalet nadal nie są powszechnie stosowane w polskim transporcie. Dzięki ich zastosowaniu firmy transportowe i spedycyjne mogą zwiększyć efektywność wykorzystania taboru samochodowego. Pozycja jest adresowana głównie do studentów logistyki i transportu, ale będzie również ciekawą lekturą dla osób pracujących już w tych branżach.

Publikację można nabyć w księgarni internetowej wydawnictwa: <http://www.ksiegarnia.difin.pl>.

