

Aleksander Szejnberg

Józef Hurek

Ryszard Gmoch

Uniwersytet Opolski, Opole

ZASTOSOWANIE TECHNOLOGII KOMPUTEROWEJ W TESTOWANIU OSIĄGNIĘĆ UCZĄCYCH SIĘ

Streszczenie: Powyższa publikacja przedstawia wyniki kilkuletnich prac nad tworzeniem spójnego pakietu programów do testowania komputerowego osiągnięć uczących się. Scharakteryzowane zostało opracowane do tej pory oprogramowanie, które podzielono na cztery grupy programów: programy testujące - weryfikujące, programy testujące z wykorzystaniem metod adaptacyjnych, programy do analizy wyników na bazie współczesnej teorii testu IRT (item response theory, teoria odpowiedzi na zadanie) oraz programy do gromadzenia i zarządzania tworzonymi zbiorami zweryfikowanych zadań testowych. Poza charakterystyką wymienionych grup przedstawiono wizję całego planowanego systemu komputerowego testowania. Zanalizowano również jego spodziewane i już zbadane w zespole zalety, w porównaniu z tradycyjnymi sposobami testowania.

Problematyka badawcza Zakładu Dydaktyki Chemii Uniwersytetu Opolskiego była w ostatnich kilku latach skupiona wokół zagadnień komputerowego wspomaganie procesu kształcenia chemicznego na różnych szczeblach nauczania. Szczególną uwagę poświęcono opracowaniu i badaniu skuteczności różnego typu zadań testowych oraz opracowywaniu narzędzi pomiaru dydaktycznego umożliwiających konwencjonalną jak i komputerową kontrolę wiedzy uczących się. Efektem tych prac była z jednej strony

empiryczna weryfikacja różnego typu zadań testowych [1-4], a z drugiej strony powstanie pakietu programów [5 - 9], wśród których można wyróżnić następujące grupy:

1. grupa programów testująco - weryfikujących
2. grupa programów do testowania adaptacyjnego
3. grupa programów do analiz wyników testowania na bazie współczesnej teorii testu IRT
4. oprogramowanie służące magazynowaniu i zarządzaniu tworzonymi zbiorami weryfikowanych zadań testowych

Ciągle rozszerzany i modyfikowany pakiet ma w przyszłości utworzyć dynamiczny system do adaptacyjnego testowania uczących się. Charakterystyka poszczególnych grup oprogramowania przedstawia się następująco:

Grupa programów testująco-weryfikujących obejmuje programy, które realizują testowanie w klasyczny sposób (zadania przydzielane są po kolei), przy czym sam test niekoniecznie musi być przeprowadzany komputerowo (to nauczyciel może wprowadzić sekwencje odpowiedzi uczniów). W wariacie komputerowym (w odróżnieniu od „papierowego”) uczeń nie ma możliwości wracania do poprzednich zadań. Programy te dokumentują odpowiedzi ucznia w plikach wyników surowych. Umożliwia to następnie szybką analizę wyników indywidualnych oraz przebadanej populacji, a także obliczenie podstawowych charakterystyk poszczególnych zadań i całego testu. Programy tej grupy to:

ZITU - program realizujący test złożony z zadań typu uzupełnienia

ZWW - program realizujący test złożony z zadań typu wielokrotnego wyboru (w szczególnym przypadku typu „prawda - fałsz”)

WOPF - program realizujący test złożony z zadań typu wielokrotnej odpowiedzi „prawda - fałsz”

ZPR - program realizujący test złożony z zadań typu przyporządkowania różnowartościowego.

A oto charakterystyka jednego z tej grupy programów realizującego test złożony z zadań wielokrotnego wyboru typu prawda - fałsz, tj. WOPF:

Utworzone oprogramowanie realizuje testowanie sekwencyjnie metodą klasyczną (wszystkie zadania po kolei). Jego celem jest zbadanie możliwości zastosowania wymienionego typu zadań w teście adaptacyjnym poprzez określenie stopnia trudności zadań i ich mocy różnicującej. Jest to problem o tyle bardziej skomplikowany, iż omawiany typ zadań testowych nie jest „zero - jedynkowy” (jak to miało miejsce w opracowanym oprogramowaniu dla testów z zadaniami wielokrotnego wyboru i uzupełnienia). Jest to podstawowa trudność w realizacji postawionego celu.

Informacje na temat interface użytkownika :

Komunikacja z użytkownikiem odbywa się za pomocą systemu menu znanego już z wcześniejszych programów. Menu składa się z następujących opcji:

- Dane* - opcja pozwala tworzyć pliki danych, odczytywać, prowadzić ich edycję (poprawianie, uzupełnianie, zmiany ...).
- Test* - opcja uruchomienia testu.
- Raport* - opcja pozwalająca obejrzeć, wydrukować lub zapisać na dysku wyniki uczniów oraz charakterystyki zadań.
- Środowisko* - opcja obejmująca pomocnicze funkcje programu (konfiguracja).
- Język* - opcja pozwalająca wybrać język w jakim odbywać się będzie komunikacja programu z użytkownikiem.
- Koniec* - pozwala zakończyć pracę z programem.

Informacje na temat przygotowania danych :

Wpisanie kolejnych zadań testu umożliwia pełnoekranowy edytor tekstowy. Zależnie od wyboru możliwa jest praca w trybie tekstowym lub graficznym. Techniczna strona pracy z edytorem objaśniona jest w „help’ie” programu. Edytor uruchamia sekwencja opcji : „dane --> odczyt ---> edycja” lub „dane --> nowe --> tekst/grafika” (po poprawnym podaniu hasła). Każde zadanie należy pisać na oddzielnym ekranie. Treść można rozmieszczać dowolnie. Istotne jest natomiast właściwe zaznaczenie miejsc na ekranie (dla kursora lub myszki), które służyć będą do wpisywania lub wskazywania odpowiedzi. Do tego celu służy znak „~” (tzw. tylda, chr(126)), po którym należy wpisać prawidłową odpowiedź. W teście typu „prawda - fałsz” jest to odpowiednio „0” lub „1”. W teście zadań „prawda - fałsz” należy pamiętać o pozostawieniu odpowiednio dużego miejsca po znakach „~0” lub „~1”, aby w trybie testowania zmieściły się w tych miejscach napisy „prawda” lub „fałsz” (w wybranej wersji językowej programu).

Informacje na temat przeprowadzenia testowania :

Po wybraniu pliku z zadaniami testu (opcje : „dane --> odczyt”) należy uruchomić opcję „test”. Po wpisaniu nazwiska i imienia ukazuje się ekran z pierwszym zadaniem. Za pomocą myszki lub klawiszami kursora można przemieszczać kursor do kolejnych pytań (stwierdzeń). Lewym klawiszem myszki bądź klawiszem *spacji* należy naciskać aż program wyświetli w danym miejscu wybraną odpowiedź. Za pomocą klawisza „PgDn” można przejść do następnego zadania testu. Test można przerwać na każdym etapie naciskając klawisz „ESC” lub prawy klawisz myszki (oczywiście program „odnotuje” to automatycznie w pliku protokołu - patrz następny akapit).

Informacje na temat analizy wyników:

Aby można było dokonać analizy wyników opracowane programy zapisują przewidziane raporty (opcja „Raport”). Ponadto automatycznie generowany jest na dysku plik z „surowymi” wynikami testu dla każdego pliku z zadaniami. Automatycznie tworzony jest także plik z raportem, w którym program zapisuje wszelkie czynności wykonywane przez użytkownika, w szczególności przebieg testów. Zapisane wyniki wykorzystać można zarówno w ocenie uczniów jak i do oceny zadań.

Celem tworzenia drugiej grupy programów służących do testowania adaptacyjnego była i nadal jest analiza różnych modeli adaptacyjnych pozwalających na szybką i skuteczniejszą (obiektywną) ocenę testowanych. Programy te wykorzystują zadania testowe wykalibrowane z wykorzystaniem programów z pierwszej grupy. Bada się między innymi korelację [8] między wynikami testowania uzyskiwanymi na drodze testowania klasycznego, a wynikami testowania uzyskanymi przy zastosowaniu programów do testowania adaptacyjnego. O ile rozbudowa pierwszej grupy programów sprowadza się do uwzględniania kolejnych typów zadań testowych, to w grupie programów adaptacyjnych badaniu podlegają kolejne modele adaptacyjne. Do tej pory opracowano oprogramowanie i prowadzono badania z trzema modelami : piramidalnym zawierającym zadania o wskaźnikach trudności w zakresie 5 - 95 % z krokiem równym 5 %, dwuetapowym oraz modelem piramidalnym, w którym jednak hierarchia zadań tworzona była w oparciu o taksonomię celów kształcenia ABC lub ABCD. Grupa druga obejmuje następujące oprogramowanie :

PIRAMIDA, TPIRAMID - programy służące do adaptacyjnego testowania w oparciu o model piramidy z wykorzystaniem odpowiednio zadań testowych typu uzupełnienia i wielokrotnego wyboru,

ANALOGCHEM, DIETAP - programy realizujące model dwuetapowego testowania adaptacyjnego z wykorzystaniem zadań odpowiednio typu ANALOGCHEM i zadań wielokrotnego wyboru,

TAXTESTI, TAXTEST2 - programy realizujące model dwuetapowego testowania adaptacyjnego w oparciu o odpowiednio trój- i czteropoziomą piramidę z wykorzystaniem zadań testowych wielokrotnego wyboru odpowiednio z kategorii taksonomicznej A,B,C lub A,B,C,D. (Kategorie: A - zapamiętanie wiadomości; B - zrozumienie wiadomości; C - zastosowanie w sytuacjach typowych; D - zastosowanie w sytuacjach problemowych.) Omawiana grupa programów będzie poszerzona , ponieważ poszukiwanie nowych i efektywnych modeli testowania stanowi istotę prowadzonych badań.

Programy do analiz wyników testowania na bazie współczesnej teorii testu IRT [10,11] - item response theory (teoria odpowiedzi na zadanie), służą do przetwarzania macierzy wyników generowanych przez programy testujące-

weryfikujące, na bazie teorii IRT. Jak dotąd nasze zadania kalibrowane były w oparciu o klasyczne charakterystyki (stopień trudności, moc różnicująca), które jednak zbyt mocno zależą od badanej populacji. Ze względu na to, że zależy nam na bardziej ścisłych kryteriach oceny trudności zadań oparliśmy ich analizę na modelu IRT (a dokładniej - na uproszczonym modelu Rascha), co pozwoli dokładniej i obiektywniej ocenić wiedzę testowanych. Do zadań tej grupy należą programy serii „LOGGITS” służące do analiz symulacyjnych, jak i do analizy rzeczywistych macierzy Rascha.

Opracowane tu niezależne procedury do ustalania trudności zadań oraz oceny testowanych współpracują dotychczas z programami grupy pierwszej. W przyszłości planowane jest ich zintegrowanie z bankiem zadań. Pozwoli to automatycznie, na bieżąco, kalibrować i uaktualniać charakterystyki zadań w bazie. Grupa programów będzie musiała być rozbudowana i zmodyfikowana, ponieważ, jak już wspomniano, dotychczas badane modele bazowały na klasycznych charakterystykach zadań.

Oprogramowanie służące magazynowaniu i zarządzaniu tworzonymi zbiorami weryfikowanych zadań testowych to w szczególności program typu „baza danych”. Jednak biorąc pod uwagę to co powiedziano wcześniej, nie możemy ograniczyć się do zwykłej bazy. Winien to być system, generujący dynamicznie różnego typu testy (wg zadanych kryteriów - modeli), umożliwiający weryfikację charakterystyk poszczególnych zadań na podstawie rejestrowanych wyników. Wtedy funkcje pierwszej grupy programów ograniczą się co najwyżej do zaledwie jednego z modeli testowania (liniowego).

W założeniach przyjęto, że bank zadań powinien być możliwie uniwersalny, tzn. obejmować zadania testowe zróżnicowanego typu, oraz umożliwiać wykorzystywanie szeregu współcześnie stosowanych charakterystyk, tak, aby dać użytkownikowi możliwie dużą swobodę w konstrukcji testów różnego typu i o pożądanym parametrach. W tym celu w banku zadań oprócz treści przechowywany jest również zbiór informacji dodatkowych, nazywany dalej atrybutami zadań (wydruk 1).

Kodowaniu i rozkodowywaniu służy dodatkowy plik „bankdef.cfg” (wydruk 2), który może być dowolnie modyfikowany i rozszerzany (uzupełniany) przez użytkownika banku zadań.

Wydruk 1. Przykładowe atrybuty zadania (z oknem wzorca odpowiedzi)

Dziedzina wiedzy.....	<i>Elektrochemia</i>
Dział.....	<i>Dysocjacja elektrolityczna</i>
Rodzaj zadania.....	<i>obliczeniowe</i>
Typ zadania.....	<i>wielokrotnego wyboru</i>
Liczba alternatyw.....	<i>4</i>
Oceniany cel.....	<i>Znajomość współczynnika dysocjacji</i>

Wzorzec odpowiedzi.....	a)..0
Grupa taksonomiczna.... <i>B</i>	b)..1
Trudność.....	c)..0
Moc różnicująca	d)..0
Atrakcyjność dystraktorów.....	
Wskaźniki IRT.....	
Autor zadania.. <i>Aleksander Szejnberg</i>	
Data ostatniej weryfikacji <i>01.01.1996</i>	
Data ostatniego wykorzystania. <i>01.01.1996</i>	
Nazwa ostatniego testu..... <i>elch_1.dta</i>	
Identyfikator zadania..... <i>eldy4_1</i>	

Wydruk 2. Przykładowa zawartość pliku konfiguracyjnego „bankdef.cfg”

- ```

#1 Dziedzina wiedzy
0 Brak przyporządkowania
1 Chemia nieorganiczna
2 Chemia organiczna
3 Biochemia
4 Elektrochemia

#2 Dział
0 Brak przyporządkowania
1 Dysocjacja elektrolityczna
2 Węglowodory

#3 Rodzaj zadania
0 Brak przyporządkowania
1 jakościowe
2 obliczeniowe
3 eksperymentalne

#4 Typ zadania
0 Brak przyporządkowania
1 wielokrotnego wyboru
2 wyboru alternatywnego prawda - fałsz
3 wielokrotnej odpowiedzi prawda-fałsz
4 wielokrotnego przyporządkowania

```

Plik „bankdef.cfg” ma zwykły tekstowy format ASCII, zatem jego modyfikacja i uzupełnianie można prowadzić dowolnym edytorem tekstu, istotne jest tylko zachowanie kolejności czterech parametrów. Ich wyliczanie rozpoczyna każdorazowo wiersz tekstu rozpoczynający się znakiem „#” ( Dalsza część tego wiersza pełni tylko rolę komentarza ). Kolejne wiersze po wyróżnionej linii zawierają dwie pozycje - kod liczbowy i przyporządkowaną mu nazwę pola, która musi być oddzielona co najmniej jedną spacją. Dopuszczalna długość każdej z czterech list wynosi 256 pozycji (rozmiar typu *byte*). Z wyjątkiem ostatniego parametru, każdy z trzech poprzednich parametrów kończy znak „#” rozpoczynający listę kodów i nazw kolejnego. Dopuszczalne są ewentualne puste wiersze odstępu, bowiem program je ignoruje.

A oto charakterystyka funkcji programu: Interfejs użytkownika programu „bankzad” przypomina wyglądem i sposobem korzystania platformy wcześniej opracowanych przez nas programów do testowania. W szczególności identyczne są hasła głównego menu (Dane, Test ,Raporty, Informacja, Pomoc, Otoczenie, Język, Koniec), jednak w przypadku trzech pierwszych przewidziano inne funkcje.

Opcja „Dane” wraz z podmenu (Odczyt, Edycja, Nowe, Hasło ) służy do wprowadzania i edycji zadań banku, ale także ich atrybutów. W programie możliwa jest zmiana tylko niektórych atrybutów, a mianowicie tych, które znane są z góry. Pozostałe z nich ( uzyskiwane dopiero w wyniku weryfikacji statystycznej, tj.: moc różnicująca, trudność i parametry IRT) będą aktualizowane przez programy testujące wykorzystujące dane zadanie. W zbiorze atrybutów przewidziano w związku z tym dane dotyczące dodatkowo daty weryfikacji i nazwy testu, na podstawie którego jej dokonano. W przypadku klasycznych parametrów : moc różnicująca i stopień trudności, przewidziano pamiętanie ich wartości uzyskanych dla pięciu populacji ( np. kolejnych pięciu lat studiów ), ponieważ miary te nie są niezależne od populacji jak parametry teorii IRT.

Opcję „Test” przewidziano do uruchomienia generowania testu w postaci listy zadań o zadanych przez użytkownika parametrach ( atrybutach ) opisanych wyżej. Niektóre z atrybutów nie będą przy tym wykorzystywane, ponieważ ich funkcja jest czysto informacyjna. Dla przykładu „Atrakcyjność dystraktorów” służyć ma w zamyśle użytkownikowi do eliminowania źle skonstruowanych wariantów odpowiedzi lub wykrywania błędów w nauczaniu - co trudno byłoby zautomatyzować, choć pomoc w tym może kolejna opcja „Raport”.

Opcja „Raport” ma służyć do udostępniania jak najobszerniejszej i syntetycznej zarazem informacji o zbiorze zadań, ich dotychczasowym wykorzystaniu i możliwości budowania testów o zadanych parametrach.

Ma również wskazywać zadania złe - o niewłaściwych wskaźnikach uzyskiwanych podczas weryfikacji. Powinno to pomóc użytkownikowi zarządzać bankiem, a także uzyskiwać informacje, których może dostarczyć tylko analiza statystyczna wyników odpowiedzi.

Informacje dotyczące zarządzania danymi przedstawiają się następująco:

W trakcie pracy z programem „bankzad” należy oczekiwać częstej ingerencji w zbiór zadań, wprowadzono więc atrybut o nazwie „Identyfikator zadania”. To on, a nie numer rekordu (który może się zmieniać) w banku, jednoznacznie identyfikuje dane zadanie. Wiąże się z tym konieczność automatycznego tworzenia, kontroli i uzupełniania pliku indeksowego o nazwie „banktask.id” zawierającego spis wykorzystanych do tej pory identyfikatorów. Chodzi o to, aby nie dublować nazw identyfikatorów zadań w bazie (to jest stosunkowo łatwe do wykrycia), ale również nie korzystać z identyfikatorów zadań, które z różnych względów zostały z bazy usunięte. W programie przewidziano stopniowe, dwuetapowe usuwanie zadań. Mianowicie zadania usuwane z zasadniczego pliku banku są jedynie przesuwane do zbioru zadań usuniętych o nazwie „deleted.bnk”; dopiero usunięcie ich z tego pliku oznacza ostateczne skasowanie. Przewiduje się również możliwość operacji odwrotnej - tzn. kopiowanie zadań do banku z pliku zadań usuniętych oraz z plików danych programów do testowania (weryfikacji).

Informacje na temat rozbudowy programu :

Projektowany bank zadań testowych będzie systemem dość złożonym i prawdopodobnie przez dłuższy czas będzie wymagał ustawicznych modyfikacji, ponieważ nie wszystkie potrzeby i sytuacje da się na etapie jego projektowania przewidzieć. W pierwszej kolejności należy zmodyfikować dotychczas opracowane przez nas programy do weryfikacji zadań testowych różnego typu, tak aby umożliwiały automatyczną aktualizację zadań i ich atrybutów w banku. Pozwoli to na wprowadzenie do banku pewnej liczby zadań o rzeczywistych atrybutach. Dopiero dysponując takim bankiem (niekoniecznie bardzo obszernym) będzie możliwe przetestowanie i dopracowanie procedur programu „bankzad”, generujących testy oraz raporty według zadanych atrybutów. Prace w tym kierunku są obecnie prowadzone, a ich rezultaty będą opisane w kolejnych publikacjach. W dalszej perspektywie prawdopodobnie konieczna będzie zmiana formatu prezentacji zadań na multimedialną, treść zadań zatem będzie musiała być przygotowywana w formacie języka HTML.

## LITERATURA

1. A.Sztejnberg, R.Gmoch, Z badań zmierzających do mikrokomputerowego doskonalenia kształcenia chemicznego na średnim szczeblu nauczania. I Weryfikacja zadań testowych służących komputerowemu sprawdzaniu umiejętności intelektualnych uczniów w zakresie chemii, Zeszyty Naukowe Uniwersytetu Opolskiego, Chemia 17, Prace dydaktyczne, 1994, 27
2. A.Sztejnberg, R.Gmoch, Z badań zmierzających do mikrokomputerowego doskonalenia kształcenia chemicznego na średnim szczeblu nauczania. II Projektowanie programów komputerowych z zakresu pomiaru dydaktycznego i charakterystyka opracowanego pakietu oprogramowania, Zeszyty Naukowe Uniwersytetu Opolskiego, Chemia 17, Prace dydaktyczne 1994, 39
3. A. Szejnberg, R.Gmoch, J.Hurek, A.Suchan, Z badań nad skutecznością różnych typów chemicznych zadań testowych. I. Prezentacja zadań testowych wyboru alternatywnego, Aktualni otázky didaktiky chemie, wyd. Gaudeamus, WSP, Hradec Králové, 1977, 101
4. A. Szejnberg, R.Gmoch, J.Hurek, A.Suchan, Z badań nad skutecznością różnych typów chemicznych zadań testowych, II. Prezentacja zadań testowych typu wielokrotnej odpowiedzi „prawda - fałsz”, aktualni otázky didaktiky chemie, wyd. Gaudeamus, WSP, Hradec Králové, 1977, 105
5. J.Hurek, A.Szejnberg, R.Gmoch, A.Suchan, Z badań nad komputerowym wspomaganie pomiaru dydaktycznego, Aktualni otázky didaktiky chemie, wyd. Gaudeamus, WSP, Hradec Králové, 1977, 112
6. R.Gmoch, J.Hurek, A.Szejnberg, Komputerowe sprawdzanie osiągnięć szkolnych uczniów. Program PIRAMIDA, Komputer w Szkole 1993, nr 9
7. R.Gmoch, A.Szejnberg, Zastosowanie komputerow w kształceniu chemicznym, Komputer w Szkole 1991, nr 10, 32
8. R.Gmoch, J.Hurek, A.Szejnberg, Z badań nad komputerowym sprawdzaniem wiedzy studentów chemii z zakresu elektrochemii przy zastosowaniu piramidalnej metody testowania adaptacyjnego, Komputer w Edukacji 1996, nr 1-2, 43
9. R. Gmoch, Wybrane problemy komputerowego wspomaganie kształcenia chemicznego, Uniwersytet Opolski, Opole 1995
10. G.H.Fischer, I.W.Molenaar (Editors), Rasch Models. Foundations, recent developments and applications, Springer- Verlag, New York, Berlin, Heidelberg, 1995
11. J.M.Linacre, Rasch measurement transactions, Part 2, MESA Press, Chicago 1996

Aleksander Szejnberg

Józef Hurek

Ryszard Gmoch

### Using computer technology for testing achievements of learners

**Abstract:** The paper presents results of several-year research on creating an uniform set of programs for computer testing of achievements of learners. The programs created up till now are characterized and divided into four groups: testing and verifying programs, programs testing by means of adaptation methods, programs for analyzing results on the ground of the contemporary IRT test theory, and programs for storing and managing sets of verified test problems. Apart from characteristics of the said groups, a vision of the whole planned computer testing system is presented. Expected and investigated in our research group advantages are compared to traditional testing methods.