Australasian Journal of Educational Technology

2010, 26(5), 585-606

Interactive computer based assessment tasks: How problem-solving process data can inform instruction

Nathan Zoanetti
The University of Melbourne

This article presents key steps in the design and analysis of a computer based problem-solving assessment featuring interactive tasks. The purpose of the assessment is to support targeted instruction for students by diagnosing strengths and weaknesses at different stages of problem-solving. The first focus of this article is the task piloting methodology, which demonstrates the relationship between process data and a priori documented problem-solving behaviours. This work culminated in the design of a *Microsoft Excel* template for data transcription named a *Temporal Evidence Map.* The second focus of this article is to illustrate how evidence from process data can be accumulated to produce and report instructionally useful information not available through traditional assessment approaches. This is demonstrated through the production of reports profiling individual student outcomes against important aspects of problem solving.

Introduction

The work described in this article is part of an ARC Linkage project undertaken in cooperation with the industry partner, North Shore Development Centre, Ltd (NSDC). NSDC is a private tuition college operating in Australia and New Zealand. The role of the broader project was to migrate an existing program of research into the assessment and instruction of problem solving (Wu, Griffin, Dulhunty & Mak, 2002; Wu & Adams, 2006) to computer based media. The rationale was that the computer could capture detailed information about student cognition which could in turn be used to better inform instruction.

Practical benefits of computer based assessment include automated and rater-free scoring, rapid feedback, and increased accessibility. Benefits related to educational measurement include the capacity to capture detailed process data and the potential to build tasks which assess skills that cannot be examined (conveniently) by other means (Mills, Potenza, Fremer & Ward, 2002; Ridgway & McCusker, 2003). Survey evidence also suggests that interactive, computer based tasks are engaging through the immediate appeal of their graphics and the sustained appeal of their interactivity (Richardson et al., 2002).

Recently, computer-based assessments that go beyond being reproductions of existing paper and pencil assessments have emerged in various domains of problem solving (Bennett, Persky, Jenkins & Weiss, 2007; Vendlinski & Stevens, 2002; Williamson et al., 2004, Masters, 2010). These assessments record detailed interactions between the problem solver and the task environment and thereby capture salient solution processes in an unobtrusive way (Bennett et al., 2007; Chung & Baker, 2003). These actions culminate in a large amount of process data that can be linked to theories of cognition and developing competence (Pelligrino, Chudowsky & Glaser, 2001; Williamson, Mislevy & Bejar, 2006). Importantly, this process data can be used to evaluate the systematicity and efficiency with which a problem solver completes tasks (Wirth & Klieme, 2003). This information makes it possible to describe *how* students solve problems rather than simply *if* they solve them. In complex domains where a variety of skills and dispositions may influence performance, knowing how students solve problems will provide more valid links to their individualised instructional needs.

The collection of process data from complex tasks presents a number of challenges to the assessment designer. First, there is a need to establish an interpretative framework for identifying meaningful performance features in potentially unwieldy strings of process data. Second, there will be a strong motivation to automate this process, which can require considerable programming expertise (Masters, 2010). Third, distinct elements of process data can describe diverse aspects of cognition, so a theoretical framework must be in place for deciphering which data can be used as evidence about which proficiencies (Mislevy, 2008). Further, the last century of test analysis has focussed on summative scoring models most often incorporating single scores per assessment task. Therefore, emerging measurement models need to be evaluated so that they can accommodate complex data, multiple observable and hypothesised variables, and support the purpose of the assessment. Why go to the extra effort? The answer is that assessments featuring complex tasks can provide cognitively diagnostic inferences unlike those typically available from traditional educational assessment instruments. This provides new information for targeting student instruction.

This section provides an overview of relevant problem-solving theory and its implications for task design. A brief description of the Cognitive Task Analysis (CTA) methodology applied in this study is provided with example tasks. The present study focused on puzzle-type problem tasks that could be solved by search-based heuristic strategies of varying sophistication. Schoenfeld (1985) described heuristics as "strategies and techniques for making progress on unfamiliar or non-standard problems; rules of thumb for effective problem solving" (p.15). This is very much the educationalist perspective of heuristic search (which is more general and less algorithmic than conceptions from cognitive science and artificial intelligence) and it is well-suited to the problem types investigated here. Treatment of the problem-solving process as a series of phases (understand, plan, try, check) is another important concept championed by the likes of Georg Polya as early as 1945 (Polya, 1945). More recently, the information processing perspective conceptualised problem solving as the interplay between representation and search (Mayer & Wittrock, 1996; Newell & Simon, 1972). Identifying strengths and weaknesses of students both preceding and during the phases of building an understanding of problems and searching for solutions to problems was considered important for increasing instructional targeting.

Two important paradigms are considered. First, the schema-driven versus search-based problem-solving paradigm as described by Gick (1986) is reviewed. Second, the expert versus novice (and similarly the gifted versus average and the good versus poor) paradigm is reviewed with particular reference to Chi, Glaser and Farr (1988) and Newell and Simon (1972). These paradigms are ubiquitously relevant across problem-solving studies. Therefore they were considered useful for divulging criteria for differentiating between tasks and problem-solving performances.

The schema-driven versus search-based paradigm tells us that the cognitive resources used by a problem solver depend on the presence or absence of so called schemas. Formally, Marshall (1995) describes a schema as follows:

> A schema is a vehicle of memory, allowing organisation of an individual's similar experiences in such a way the individual (1) can easily recognise additional experiences that are also similar, discriminating between these and ones that are dissimilar; (2) can access a generic framework that contains the essential elements of all of these similar experiences, including verbal and nonverbal components; (3) can draw inferences, make estimates, create goals and develop plans using the framework; and (4) can utilise skills, procedures, or rules as needed when faced with a problem for which this particular framework is relevant (p. 39).

Useful schemas are not always available to problem solvers. In these situations, an alternative problem-solving approach must be invoked. This is where search-based problem solving becomes relevant (Gick, 1986). Search-based problem solving is characterised by the application of search-based heuristic strategies that are not necessarily specific to a particular class of problems. They are usually more general, less direct and in some cases offer no guarantee of success. While algorithmic schemas for solving various puzzle tasks do exist (Luchins, 1942), search-based strategies are also sufficient for reaching the goal.

Studies of expert versus novice problem solvers have identified several differentiating aspects of performance. These differences can be used by assessment designers to identify important performance features that tasks should elicit (Glaser, 1991). Data describing actions and latencies have been used previously to explore problem-solving processes (Lohman & Ippel, 1993, p. 56). In a computer-based setting, these data can include clicks, drags, drops, mouse rollovers, mouse hold-downs, and many more. Temporal information about these interactions can also be recorded with ease. If these data provide evidence about salient performance features on the domain, then their collection should be considered a goal for assessment design. The following paragraphs provide the theoretical basis for making use of the different types of readily available process data.

Expert problem solvers are argued to be more forward-working and goal-directed (Newell & Simon, 1972), have strong metacognitive and self monitoring skills (Chi, Glaser & Farr, 1988), analyse problems qualitatively in detail and check the products of their problem solving (Gerace, 2001). Their efforts often begin with an investment in planning and analysis. Planning and analysis has been argued to help experts transcend surface features of problems not strongly related to goal attainment (Sternberg & Ben-Zeev, 2001). Instead, experts look for deep structural representations of problems prior to engaging in search (Chi, Feltovich, & Glaser, 1981). It follows from the importance of problem representations that experts will tend to get off to a relatively goal-directed start. The capacity to record the relative goal-directedness of the initial interactions of a problem solver is arguably worthwhile and could be taken as evidence about the quality of initial problem representations.

Novices may tend to focus on surface features of problems, which in this context might involve manipulating an object without having the problem goal in mind. Consequently, novice problem solvers will tend to be less efficient and their solutions will feature more redundant and erroneous interactions. They will also have a lower likelihood of successfully solving problems. These expert-novice differences reveal performance indicators which differentiate between more and less expert problem-solving approaches. Therefore these are the indicators which tasks should be designed to elicit.

The study

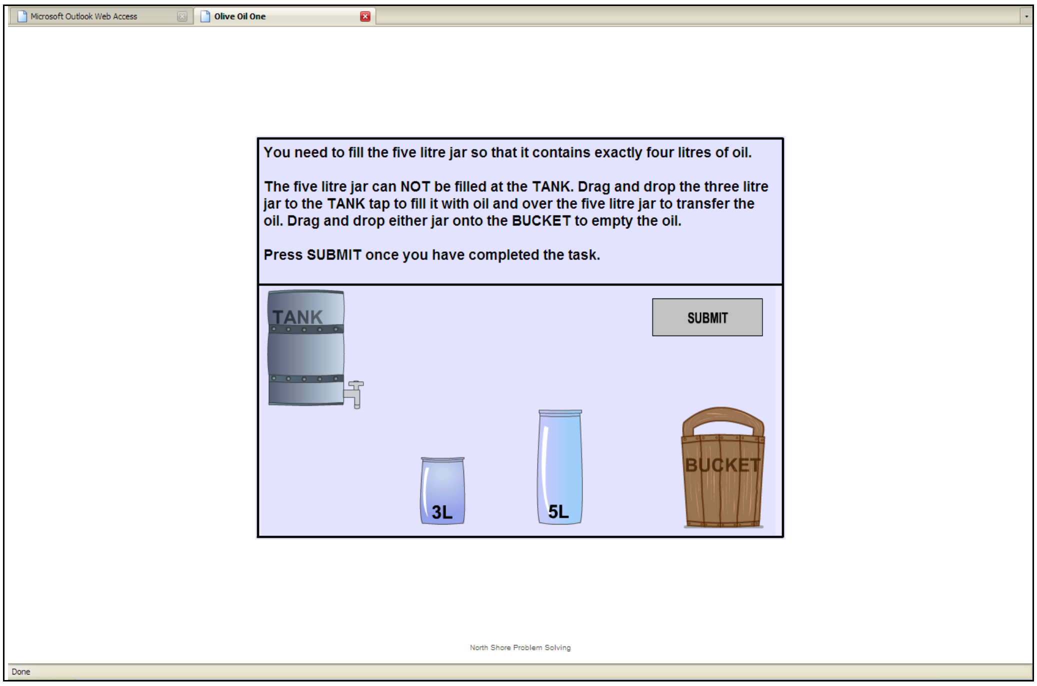
This study is divided into two parts. The first part describes a method of recording and interpreting data from the interactions of students attempting interactive computer-based problem-solving tasks. The methodology extends existing approaches by introducing an evidence transcription tool that allows experts to visualise student solution paths as they occurred as a function of time. Data interpretation is recognised as one of the cornerstones of valid assessment (Pelligrino et al., 2001). The second part of this article provides an overview of the analysis and reporting framework used to generate and communicate assessment inferences. In this study, *Bayesian Inference Networks*, or *Bayes Nets*, were employed for classifying student performances. This article concludes with an example of a student report produced in this context and an explanation of its links to targeted instruction.

The temporal characteristics of solution-path behaviour have been studied in a number of settings to reveal various inferences about cognition (Lohman & Ippel, 1993; R. J. Mislevy, 1993). Where inferences about student proficiencies are difficult to disentangle, the addition of response time information can provide some evidentiary weight (Gvozdenko & Chambers, 2007). Usually expert search is more automatic as a result of the deeper representations. This suggests that response time, perhaps in conjunction with other information such as whether the goal was attained, could be used as an indicator of expertise.

In addition to data describing the total response time for a task, several researchers have gone one level of detail further. Distinct latencies corresponding to important points within the problem (like an impasse) or distinct processes (like planning or reviewing) have received attention (Fum & Del Missier, 2001). Paek (2002) for instance found that better performing students would invest more time on the initial step of multi-step mathematics items. Establishing inferences about the adequacy of decoding time prior to task interaction was seen as a possible target inference with its own instructional implications. This is consistent with assertions by Glaser (1991), who stated that while experts tended to solve problems faster than novices, they tended to dedicate a disproportionate amount of time decoding the problem and forming a representation. Temporal information as a source of evidence about problem-solving processes is well-placed with computer-based data capture and expert-novice theories which provide a framework for interpretation. Therefore in this study, in addition to recording task response time, tasks were designed so that the latency preceding the first task interaction was also recorded.

1. Recording and interpreting data

The tasks were constructed so that objects could be clicked, dragged and dropped using the mouse cursor. Examples of tasks from this study are provided in Figure 1 and Figure 2. There was no requirement to use the keyboard. Instructions and constraints were described above the graphical task objects such that all of the information necessary to complete a task was embedded within the presented material. This is consistent with the idea of domain-general problem-solving, where prior knowledge specific to a given domain is not a prerequisite for being able to solve the problem.

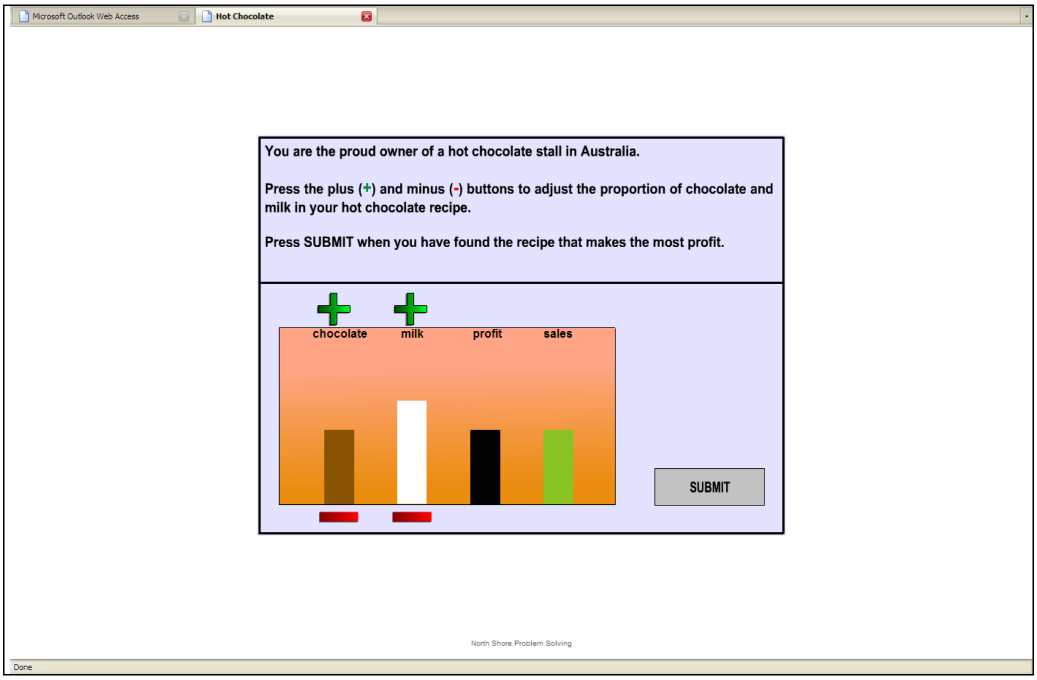


You need to fill the five litre jar so that it contains exactly four litres of oil.

The five litre jar can NOT be filled at the TANK. Drag and drop the three litre jar to the TANK tap to fill it with oil and over the five litre jar to transfer the oil. Drag and drop either jar onto the BUCKET to empty the oil.

Press SUBMIT once you have completed the task.

Figure 1: Olive oil task



You are the proud owner of a hot chocolate stall in Australia.

Press the plus (+) and minus (-) buttons to adjust the proportion of chocolate and milk in your hot chocolate recipe.

Press SUBMIT when you have found the recipe that makes the most profit.

Figure 2: Hot chocolate task

In summarising the implications of theory and previous empirical work for task construction, at least five types of performance evidence were identified as relevant. These included the correctness of solution, the presence of errors, the presence of repetition and redundancy, the total time taken, and finally, other temporal latencies. Observable variables summarising each of these indicators were specified and these are revisited in Table 3. The extent to which computer-based interactive tasks facilitated collection of data describing each of these indicators was evaluated through task piloting. This is described in the following sections.

Cognitive task analysis

From computer-captured performance data it is possible to analyse in great detail the processes and products of problem solvers' interactions with tasks (Bennett et al., 2007). It is also possible to collect this data (known sometimes as *click-stream* data) concurrently with verbal reports provided by problem solvers during task interaction (Chung, de Vries, Cheak, Stevens & Bewley, 2002). This type of evaluation is commonly referred to as *Cognitive Task Analysis* (CTA) and is typically carried out early in the assessment design process (Mislevy, Steinberg, Breyer, Almond & Johnson, 1999). The function of CTA is to validate the dependency of elicited behaviours upon various structural features of tasks and the knowledge and skills of people undertaking the tasks (Williamson et al., 2004). Importantly in this context, a CTA will reveal important relationships between potentially complex sequences of computer-captured data and corresponding observable variables. In other words, a CTA can elucidate how assessment data will be identified and evaluated as observable evidence from which student proficiencies can then be inferred (Williamson, Bejar & Mislevy, 2006).

Task piloting

The CTA was designed to facilitate the formalisation of evaluative rules for converting computer-captured process data into values for observable variables. These observables would later be accumulated into a measurement model for generating inferences about student proficiencies. The CTA would also reveal the extent to which the range of elicited patterns of behaviour was consistent with the range of documented differences between expert and novice problem solvers as defined previously. Further, where tasks were not useful for differentiating between different problem-solving approaches and capabilities, task refinements or omissions could be made prior to subsequent assessments.

A total of nine tasks were developed and piloted in this study. The tasks were designed using *Macromedia Flash 8* and were rendered within HTML web pages. Interactions with the tasks were recorded using *Actionscript 2.0* syntax and variables. Timestamps were recorded for each interaction datum so that latency data could be analysed at a later stage. At the conclusion of the tasks, the variables were posted to and processed by a PHP script which submitted the student interaction record (sometimes referred to as a *work product* as shown in Figure 3) to a *MySQL* database for post-hoc analysis. Future directions for this work are likely to involve real-time data analysis to facilitate adaptivity in the selection of tasks.

Data collection

Three sources of process evidence were collected; computer click-stream data (describing key strokes and mouse actions), verbal commentary and occasional behavioural observations (such as an expression of relief or frustration). It was intended that these three sources of evidence be combined for maximum interpretability as per Chung *et al.* (2002). The interpretation of the computer-based evidence was supplemented by supporting verbal and behavioural evidence. The use of multiple sources of evidence has the advantage that sources of error and incompleteness for one data form are not necessarily common across the other data forms. It is also conceivable that certain data forms offer more vivid clues about student cognition at different stages of the problem-solving process. One example from this study involved a student verbally declaring "I don't really get this" approximately 45 seconds into their solution attempt. When the student made this statement, they were yet to manipulate any objects within the task. Arguably statements such as these provide relatively unambiguous insight into a student's understanding of the problem, at least more so than their click-stream data. This is especially true since a lack of interaction at the commencement of problem solving could reflect that a problem solver does not understand how to interact with the task (such as in this example) or it could equally represent an expert investing time in planning and analysis. Relying on verbal or action protocols in isolation is arguably not as reliable as having them concurrently available.

A total of 43 students participated in the CTA study. These students were enrolled in weekend tuition programs in one New South Wales branch and two Victorian branches of NSDC. The target grade level for the CTA was grade 5 and grade 6 (typically aged 10 and 11). This age level has been identified as a target age that is sensitive to problem-solving instruction. Students were of varied abilities, given the role of NSDC in both remediating and in providing coaching for scholarship examinations. Students, with permission from their parents, volunteered to partake in the CTA. Approximately equal numbers of girls and boys took part.

A digital voice recorder was used to record the interview process. These discussions constituted what is commonly referred to as verbal protocol (Ericsson & Simon, 1993). Verbal protocol, sometimes coined the "…thinking aloud procedure…" (Webb, 1979, p.84) refers to the elicitation of processes, both productive and reflective, by a problem solver by way of verbal extraction. Verbal protocol has underpinned problem-solving research in a variety of settings (Chi et al., 1988; Schoenfeld, 1985; van der Linden, Sonnentag, Frese & van Dyck, 2001, Leighton and Gierl, 2007). In short, it is reasonable to consider verbal reports as valid (though incomplete) data about cognitive processing so long as the interviews are constructed to avoid certain pitfalls (Ericsson & Simon, 1993).

In the present study, tasks were introduced to students with the instruction that they should 'think aloud' during the problem-solving process; they were asked to explain their thoughts about the problem and its goal, how they were going to solve it, and what they were considering doing and actually doing whilst solving it. The target population was relatively young, and it was thought that this might have contributed to not all students being able to sustain the 'think aloud' protocol during the entire task without some prompting. A number of generic verbal prompts were used in an attempt to overcome this issue, paraphrasing those described in Lawson and Chinnappan (1994, p. 67). These included "keep telling me what you are doing" and "can you tell me what you are thinking about" among others. Such prompts were used if the student proceeded to manipulate task objects without explanation or if they fell silent for a period of time exceeding circa 30 seconds. For students unable to engage in the verbal protocol analysis without continuous generic prompting, more specific prompts were used to probe the level of understanding and intent of the student whenever such information was missing. This was useful for gauging the student's familiarity with the task and their understanding of the task objective and the rules for object manipulation.

The focal source of process evidence, collected in sync with the verbal protocol and behavioural observations, was the computer-captured data. This came in the form of a delimited string of data describing distinct student-task interactions each with a corresponding timestamp. This was recorded as the interaction sequence ('sq') within the work product. Other summary data were also prefixed to this sequence such as the item identifier ('it'), the submitted answer ('res') and the total response time ('rt'). In the example in Figure 3, the work product indicates the following: the student attempted task '0099' (which is the *Olive Oil* task from Figure 1); the student submitted a response of 4 litres, which happens to be the correct solution; the student spent 140.564 seconds engaged in their solution attempt; and as an example of one particular interaction, after 81 seconds the student filled the three litre jug ('81_f3') a second time. It is also apparent that the work product is not a user-friendly data representation from which to make interpretations about problem-solving processes. Therefore it was reformatted as described in the following paragraphs.

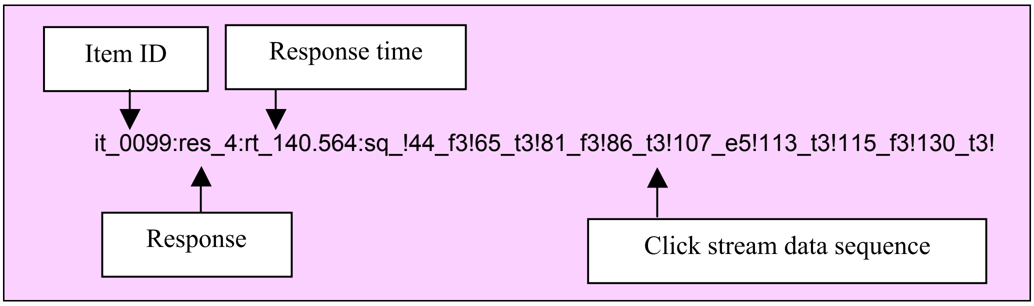

Item ID
Response time
it_0099:res_4:rt_140.564:sq_!44_f3!65_t3!81_f3!86_t3!107_e5!113_t3!115_f3!130_t3!
Response
Click stream data sequence

Figure 3: Work product comprising summary data and click stream data

All three sources of concurrent evidence were transcribed onto a *Microsoft Excel* worksheet template to produce what has been named a *Temporal Evidence Map* (TEM). The map consists of several parts but essentially links two dimensions - time and activity. The horizontal axis on each map represents time in one-second intervals. The evidence sources were transcribed together. Performances by different students were mapped in adjacent row groups to allow visual comparison and to help distinguish the differential features of performances. Presentation of concurrent process data has been demonstrated elsewhere (Chung et al., 2002; Goldman et al., 1999). However, these existing approaches have listed the different sources of process data within separated columns within conventional tabular displays.

The key difference between the proposed method and existing methods is the explicit visual representation of the student-task interaction data on a time axis. This approach preserves the interval property of time measures rather than compressing the data transcript into tabular form. With several researchers reporting discernable relationships between within-task latencies, overall response times and problem-solving processes (Marshall, 1995; Paek, 2002; Schnipke & Scrams, 1999), the importance of temporal information in the data interpretation phase should not be undermined. This study therefore presents a data transcript that upholds the importance of temporal information more so than previously demonstrated tabulation methods. Of course, what remains to be seen in follow up studies is whether other subject matter experts find the visual transcription easier to work with than tabulations, whether there is higher inter-rater reliability between experts assigning codes to TEMs, and whether the visual transcript leads to evidence identification that was not achieved using other methods.

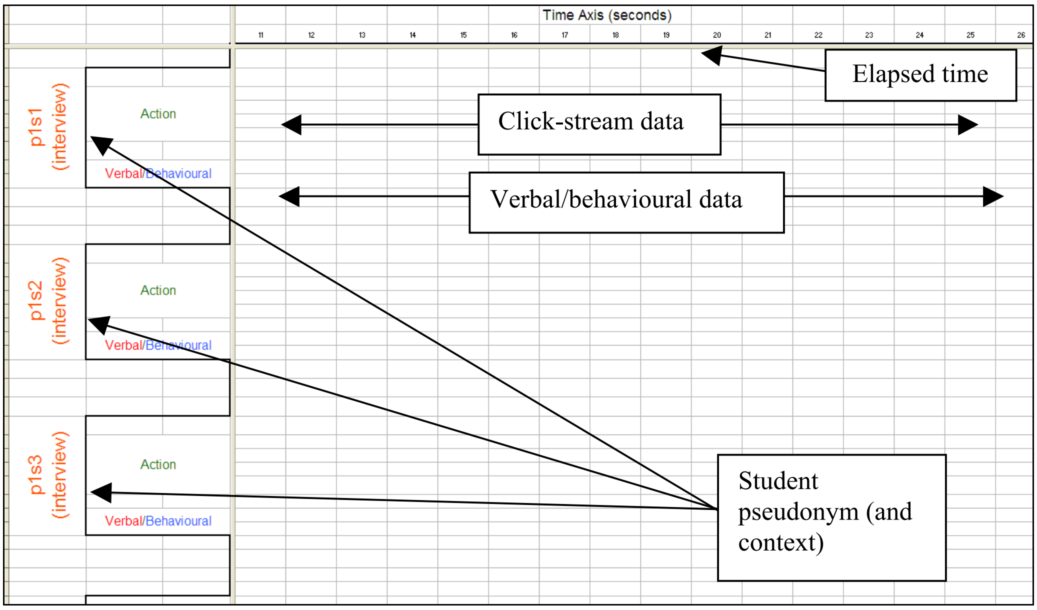


Figure 4: Empty template of a *temporal evidence map* prior to evidence transcription

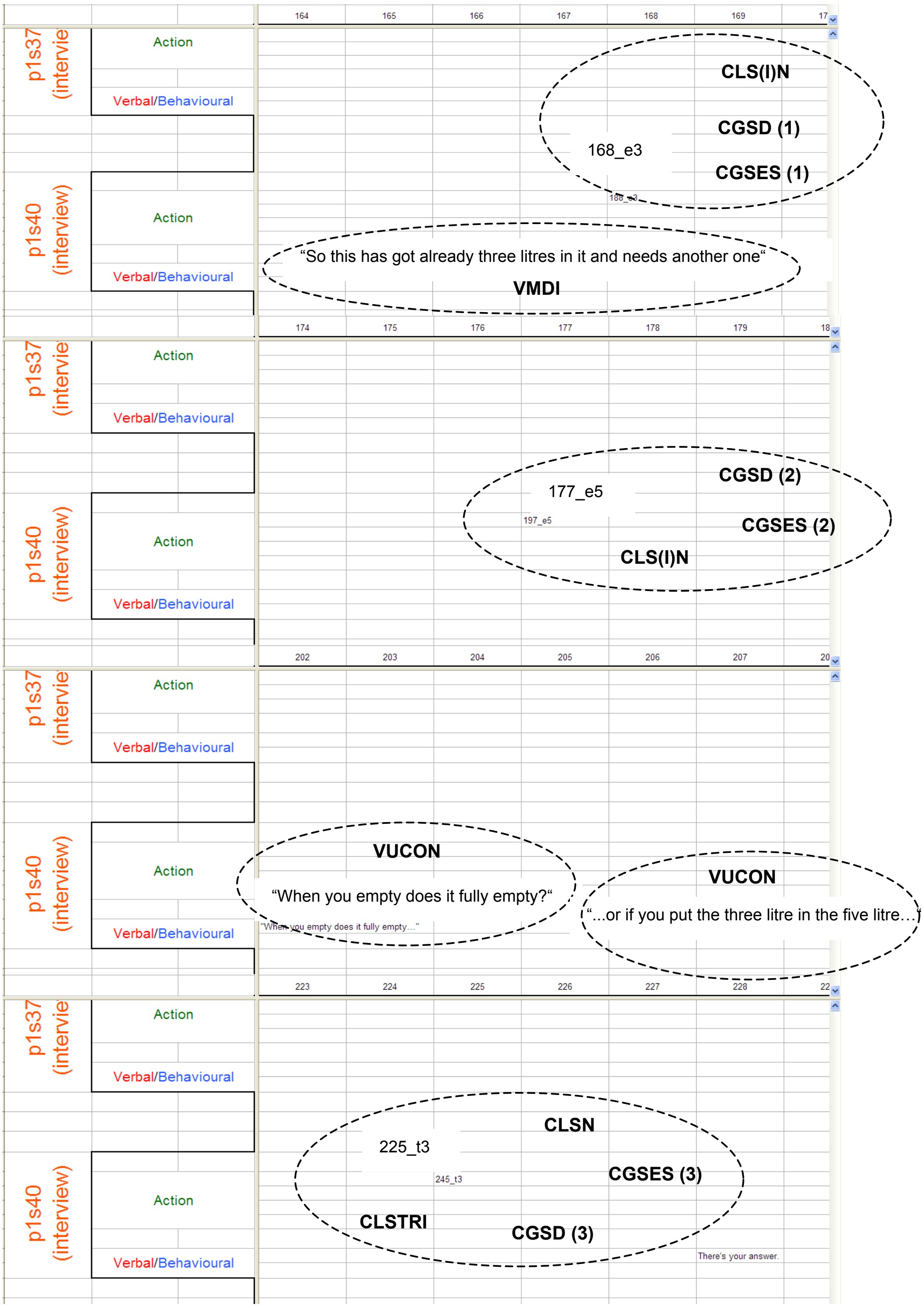
Data interpretation

Data interpretation involved the assignment of codes representing observable behaviours to raw process data. For example, an important behaviour useful for differentiating between more and less expert-like performances is whether or not problem solvers repeat ineffective actions (van der Linden et al., 2001). In such cases the data would manifest as a repeat of a particular click-stream sequence that was unsuccessful in resolving a previously-encountered impasse. The a priori expected behaviours were consolidated with code lists published by Lawson and Chinnappan (1994, p. 72), Montague and Applegate (1993, p. 31) and Flaherty (1975). The resultant set of codes is presented in Table 1.

TEMs were analysed to formalise the logical rules for assigning codes to performance features such as those described in the previous paragraph. Construction of these rules mirrored work by Bennett *et al.* (2007) for evaluating empirical data in order to develop provisional scoring rules. The assignment of codes (from Table 1) considered the following data as displayed on each TEM: immediately and objectively interpretable verbal statements and explanations; instances where verbal/behavioural evidence directly explained corresponding click-stream actions; instances where earlier/later click-stream actions are explained by verbal/behavioural evidence; instances where earlier/ later click-stream actions are explained by earlier/ later click-stream actions

Table 1: Codes for a-priori expected observable behaviours

Category			Code
Underst-anding and represent-ation	i	Reads beginning of question stem	VU1
	ii	Reads entire question stem	VU2
	iii	Paraphrases part of question stem	VU3
	iv	Paraphrases all of question stem	VU4
	v	Rereads part or all of the question after some trials	VU5
	vi	States familiarity with permutations	VF1
	vii	Reads goal verbatim	VR1
	viii	Paraphrases goal correctly	VR2
	ix	Paraphrases goal incorrectly	VR3
	x	Paraphrases goal in terms of task objects	VR4
	xi	Verbalises personal experience	VP1
	xii	Indicates task objects physically	VB1
	xiii	States understanding, e.g. “I get it”	VUY
	xiv	States lack of understanding, e.g. “I don’t really get this”	VUND
	xv	Verbalises products with uncertainty	VUNC
	xvi	Conjectures possible outcomes	VUCON
	xvii	Summarises observed consequence of action	VSUM
Familiaris-ation by task object manip-ulation	i	Initially drags any object (incomplete or non-goal directed)	CD1
	ii	Initially clicks on any object (incomplete or non-goal directed)	CC1
	iii	Repeats initial drag	CD2
	iv	Repeats initial click	CC2
	v	Continues non-goal directed actions	CCDR
	vi	Repeats and extends initial drag or click to completion	CCDE
Search distinct events	i	Generates initial trial	CLSI
	ii	Best trial first	CLSBF
	iii	Trials planned as sets	CLSTPS
	iv	Ineffective trial	CLSIN
	v	Repeats an ineffective trial	CLSR
	vi	Generates new trial (action)	CLSN
	vii	Generates new trial (verbal – e.g. “so now try”, “maybe if”)	VLSN
	viii	Guessing (random trial)	CLSG
	ix	Uncertainty (back and forth movements) within task	CLSUN
	x	Guessing (verbal)	VLSG
	xi	Confirmatory trials near goal region	CCNG
	xii	Repeated trials not near goal region	CLSNNG
	xiii	Risk-taking at an impasse	CLSTRI
Search overall	i	Cannot generate trial	CGS0
	ii	Repeated errors	CGSR
	iii	Trial diversification	CGSD
	iv	Eliminate all possibilities	CGSEA
	v	Eliminate possibilities sequentially	CGSES
	vi	Trials spatially disordered	CGSDIS
	vii	Zero redundant trials – flawless	CGSF
	viii	Search incomplete	CGSI
Emotion	i	Exclamation or movement indicating realisation	BE1
	ii	Sound indicating frustration or disappointment	VE
Meta-cognitive, reflective and evaluative	i	Recalls and explains procedure	VMRP
	ii	Evaluates trial against constraints	VMTAC
	iii	Evaluates goal against constraints	VMGAC
	iv	Reflects on an action until the next best action is planned	CMP
	v	States answer upon completion	VMA
	vi	Confirmatory trials declared (“just checking again”)	VCNG
	vii	Declaring an impasse	VMDI
	viii	Creates a strategy	VMCS



“So this has got already three litres in it and needs another one“

“When you empty does it fully empty?“

“...or if you put the three litre in the five litre…”

“When you empty does it fully empty...”

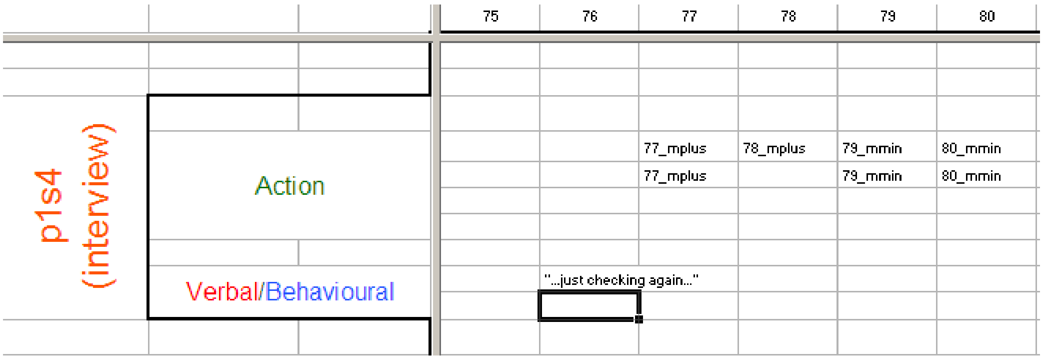
There's your answer.

Figure 5: *Temporal evidence map* segments showing a successful trial and error solution path

and verbal/behavioural evidence; instances where an accumulation of click-stream actions and verbal/behavioural evidence constituted an overall heuristic strategy; existence of key differences across problem solvers in terms of click-stream action density and click-stream action type as a function of time. It is important to note at this point that the verbal and behavioural evidence would not be used beyond the CTA. Collecting these data would not be feasible given the intended development of a standalone assessment which could be administered to students directly via computers without any assistance. Nonetheless, throughout the piloting phase, these data were considered valuable for verifying the emergence of certain problem-solving behaviours and for calibrating rules for interpreting click-stream data.

Electronically maintained examples of the TEM tool are provided in Figures 5 and 6. An example of a printed TEM is provided for illustration only in Figure 7. Printed TEMs can be spread across the length of a desk and coded with a pen. This approach was found to be much easier than coding on the computer and provided a better overview of the entire solution path for long interaction sequences.

In Figure 5, the inter-relatedness of distinct evidence elements is apparent. Conducting the third unique trial following an impasse can be classified as an implementation of trial diversification (CGSD), conducted sequentially (CGSES). This logical argument accounting for the presence, absence and order of particular evidence elements can be formalised as a general evidence interpretation rule. It is therefore included in Table 1. The performance of the student on the *Olive Oil* task depicted in Figure 5 offers a vivid example of search-based problem solving in the context of an unfamiliar task. This is in contrast to schema-driven problem solving, such as the application of a learned deterministic strategy or algorithm.



		75	76	77	78	79	80
p1s4 (interview)	Action			77_mplus	78_mplus	79_mmin	80_mmin
				77_mplus		79_mmin	80_mmin
	Verbal/Behavioural		"...just checking again..."				

Figure 6: *Temporal evidence map* segment for a student carrying out confirmatory trials while solving the *Hot Chocolate* task

Figure 6 illustrates one example of how concurrent evidence supports calibration of computer-captured process data. Once it is known what the intended strategy of a problem solver is, as explicitly verbalised, the corresponding click-stream data can be related to this strategy and evidence identification rules can be formulated or refined. In other words, once we know that a problem solver has a certain level of understanding (internal representation of a problem task) or an intention to carry out a particular strategy (this example), we can inspect the nature and rapidity of their subsequent actions to formulate our own understanding of how aspects of performance manifest in data. Such trends should constitute important discussion points between subject-matter experts and assessment design teams. It is in this way that a TEM can contribute to building an interpretive framework for click-stream data.

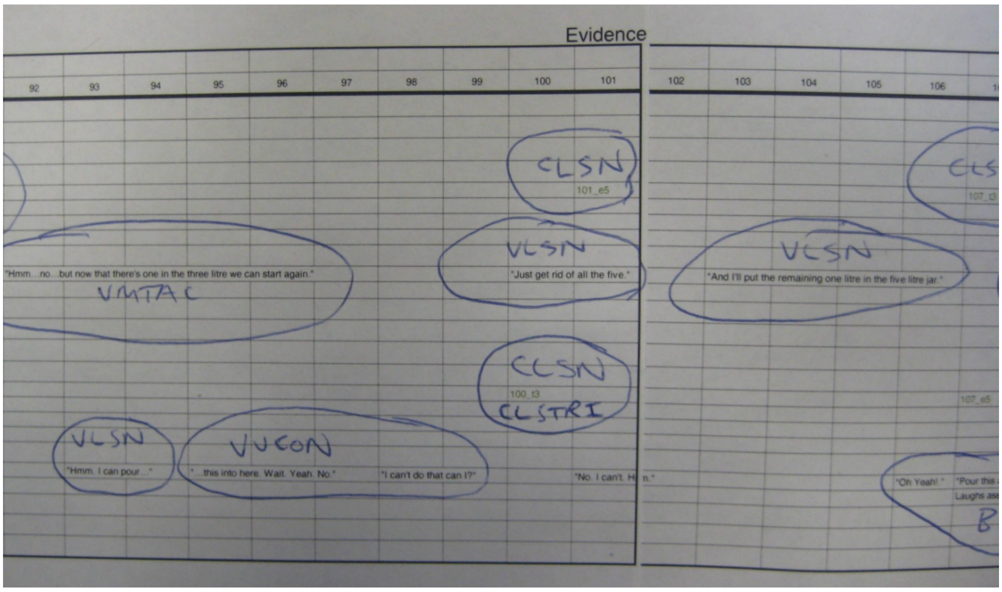

Evidence
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
CLSN
101_e5
VLSN
"Just get rid of all the five."
"Hmm...no...but now that there's one in the three litre we can start again."
VMTAC
VLSN
"And I'll put the remaining one litre in the five litre jar."
CLSN
100_t3
CLSTRI
107_t3
107_e5
VLSN
"Hmm. I can pour..."
VUCON
"...this into here. Wait. Yeah. No."
"I can't do that can I?"
"No. I can't. H n."
"Oh Yeah!"
B

Figure 7: Image of a printed TEM with scribed codes

Perusal of the TEM segments across the full range of pilot tasks revealed empirical information about the target variables identified via literature review. These are elucidated in Table 2 with relevant references from research concerning expert-novice differences.

Table 2: Profile variables identified through CTA and literature review

Profile variable	CTA empirical manifestation	Theoretical support
Decoding time	Some students tended to manipulate task objects prior to having completely read or understood the problem stem. Most students read question stems to the point of understanding at least how to manipulate objects and often related this to the goal. Other students read the problem stem at least once and if the problem was not completely understood they tended not to interact with the task for an extended period. *Categorisations:* Brief Intermediate Extended	Students should minimally spend enough time reading the problem instructions to be able to paraphrase the goal and the conditions for valid search through the problem space. Research in physics problem solving by Larkin and colleagues (Larkin, 1980; Larkin, McDermott, Simon & Simon, 1980) found that experts would devote a disproportionate amount of time to reflecting on the nature of the problem prior to engaging in solution steps. A similar finding is reported by Paek (2002) in the context of mathematics problems, where students who successfully solved problems were found on average to spend 35% more time on the first step of the problem. These findings are consistent with the consensus that good problem solvers use extensive analysis and planning prior to engaging in problem solving whereas novices might plunge into exploration prematurely (Glaser, 1991).

Initial representation	Some students tended to start searching with valid and goal-directed actions. Other students did not start problems well. Student starting problems poorly tended not to understand the problem constraints or its goal and tended to engage in irrelevant interactions with task objects. *Categorisations:* Deep Surface	Understanding a problem prior to search is a highly important stage within the problem-solving process (Polya, 1957). As Wu (2003, p. 112) explained, "…one cannot proceed with solving a problem until one understands the required task". That is, at least, without having to resort to extremely rudimentary trial and error. The importance and effort placed on understanding the problem is yet another differentiating factor between good and poor problem solvers (Whimbey & Lochhead, 1991). Experts attempt to construct deep structural representations of problems whereas novices tend to engage with surface features (Chi et al., 1981).
Error tendency	Some students performed actions that were not allowed as explained in the question stem or implicit in the task structure. Others tended to avoid invalid moves. Prolific errors across tasks were not common. *Categorisations:* Minimal Some	Error avoidance and error management have been described as critical skills in self-regulated problem solving (van der Linden et al., 2001). Frese (1995) explained that the negative emotional consequence of committing errors may compromise the cognitive resources that remain available for solving the problem.
Attainment	Some students, irrespective of search quality, persisted until they reached the goal. Others abandoned problems prior to reaching the goal. Other students reached an erroneously conceived goal. *Categorisations:* High Low	Being able to solve problems must be seen as a primary educational outcome, with the efficacy of the search being a complementary educational goal. Including an overall indication of the tendency to attain solutions can be informative when coupled with other diagnostic information. This could disambiguate the type of problem solver. For example, high attainment coupled with high speed might indicate familiarity or expertise. Low attainment coupled with high speed is more likely to indicate a lack of planning or disengaged behaviour (Stevens & Thadani, 2006).
Activity	Some students conducted too few actions to explore the problem. Others conducted many redundant actions. Some students tended to solve problems with little redundancy. *Categorisations:* Reduced Efficient Redundant	Stevens and Palacio-Cayetano (2003) identified four main strategy type classifications in terms of relevance and scope of search. To recap, experts tended to focus their search on relevant or goal-directed information. Novice-like strategies featured more redundancy, or at the other extreme, an inadequate number of actions given the scope of the problem. Consistent with this, Ahonniska *et al.* (2000) revealed that in the well-known *Towers of Hanoi* task, the lengths of pauses (in seconds) before certain critical moves were made was disproportionately higher among good performers. This would indicate the use of planning to overcome impasses rather than engaging in search and processing of a more general and less purposeful nature. Stevens and Thadani (2007) illustrate how classifying search type in this way can inform distinct instructional interventions.

Search duration	Some students abandoned problems prematurely. Other students would spend longer on problems than others either when they were stuck or when they were being careful. Categorisations: Limited Intermediate Extended	Response time measures can provide insight into the level of automaticity with which an individual is performing (Gvozdenko & Chambers, 2007; R. J. Mislevy, 1993). To elaborate, "correct and fast" could be interpreted as an insightful solution, or an indication of familiarity or expertise (Glaser, 1991). This is consistent with the finding that experts tend to work faster once they have internally represented a problem, as reported by Wankat and Oreovicz (1993). According to Sternberg (1985), time allocation during problem solving is an important indicator of information processing and persistence, and involvement in problems is highly correlated with success in solution. "Incorrect and fast" may point to a lack of persistence or a misconception of the problem.

The result of the TEM implementation was a framework of evidence rules. For each code to be carried forward to the standalone assessment version described later, evidence rules were specified (refer to Table 3).

Table 3: Evidence rule framework

Observable variable	Evidence rule structure	Profile variable

Pre-search latency (PSL)	IF PSL<X seconds THEN *low*
	IF X≤PSL≤Y THEN *intermediate*
	IF PSL>Y THEN *high*
		Decoding time

Best-first search (BFS)	*Yes*: First action/s is/are the most goal-directed *No*: Otherwise	Initial representation
Valid-first search (VFS)	*Yes*: First action complies with task instructions *No*: Otherwise	Initial representation
Goal attained (X)	*Yes*: Correct *No*: Incorrect	Attainment
Search scope	*Adequate*: Searches or interacts with at least the minimum required number of problem states or task objects for goal-attainment *Inadequate*: Otherwise	Activity

Response time (RT)	IF RT<X seconds THEN *low*
	IF X≤RT≤Y THEN *intermediate*
	IF RT>Y THEN *high*
		Search duration

Action count (AC)	IF AC<X THEN *limited*
	IF X≤AC≤Y THEN *efficient*
	IF AC>Y THEN *redundant*
		Activity

Repeated action count (RAC)	IF IAC=0 THEN *none*
	IF IAC>0 THEN *some*
		Activity

Invalid action count (IAC)	IF IAC=0 THEN *none*
	IF IAC=1 THEN *one*
	IF IAC>1 THEN *some*
		Error tendency

Following the CTA, several tasks were determined to be unsuitable for measuring search-based problem-solving characteristics. In some cases this was due to construct-irrelevant sources of task difficulty such as unnecessarily difficult calculations. In other cases, the tasks did not provide enough opportunities to differentiate between more and less expert performers, usually owing to elicitation of a limited range of actions regardless of student proficiency.

Upon completion of the CTA, frameworks were developed for constructing appropriate tasks and for identifying and evaluating performance features in the data. The empirical prominence of the problem solving behaviours inventoried in Table 1 also provided evidence of construct validity for the initial pool of tasks. The following part of this article provides a brief description of how a subsequent large-scale trial was conducted to demonstrate production of student reports for linking assessment inferences to student instruction.

2. Analysis and reporting framework

Following the piloting, several tasks were retained and new tasks were designed to produce a set of 12 problems. These were administered on a larger scale to 914 students from grades 3 to 8 responding to between 8 and 12 tasks. To illustrate how the task piloting supported the eventual analysis of assessment data, the process has been depicted in Figure 8.

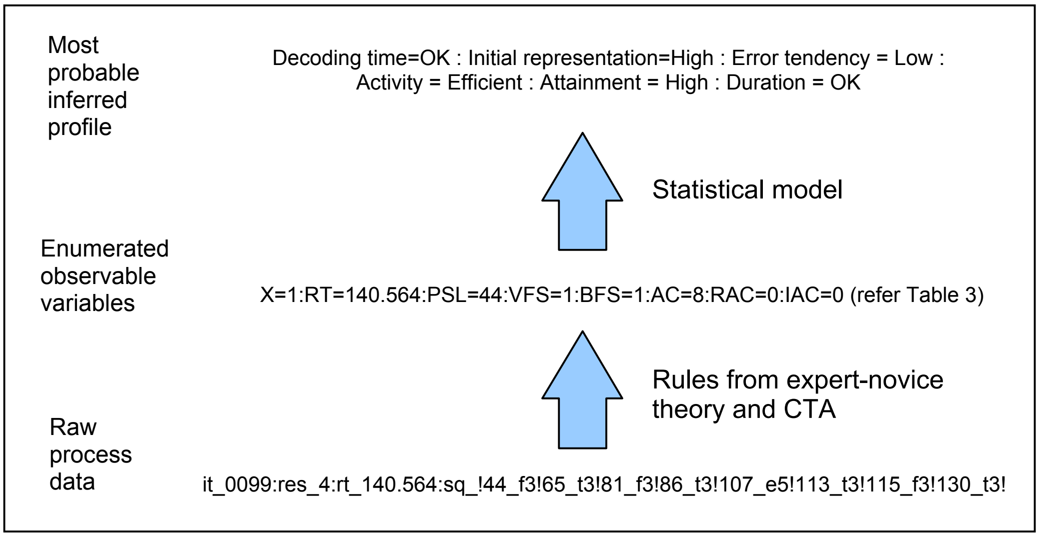
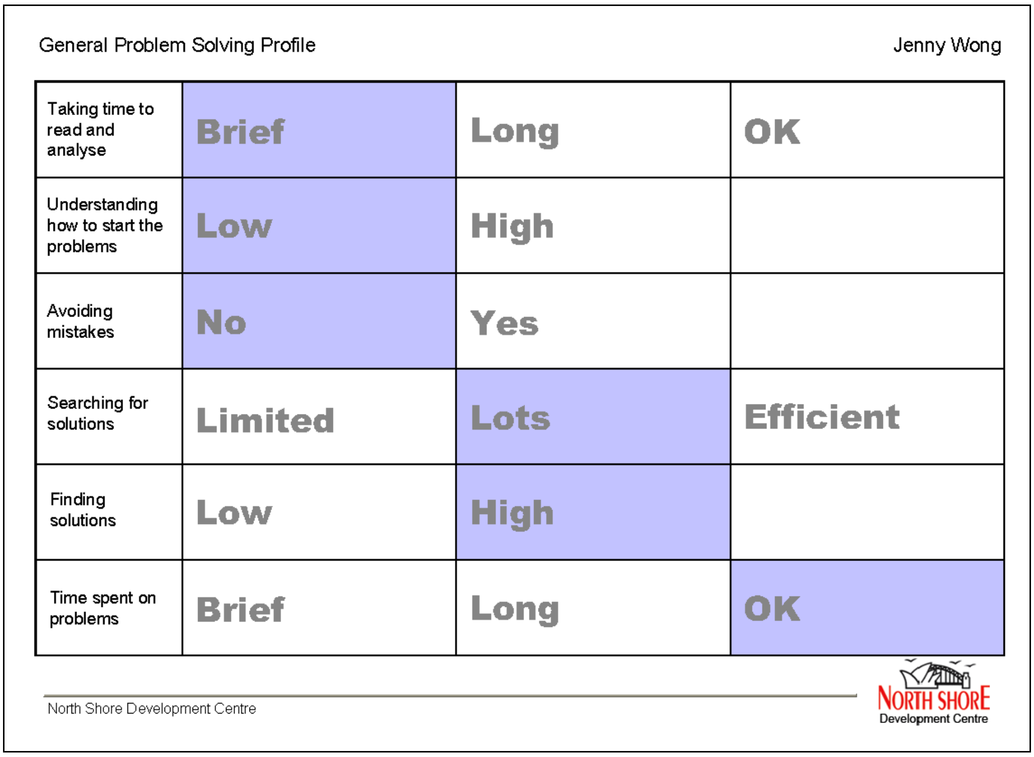

Most probable inferred profile
Decoding time=OK : Initial representation=High : Error tendency = Low : Activity = Efficient : Attainment = High : Duration = OK
Statistical model
Enumerated observable variables
X=1:RT=140.564:PSL=44:VFS=1:BFS=1:AC=8:RAC=0:IAC=0 (refer Table 3)
Rules from expert-novice theory and CTA
Raw process data
it_0099:res_4:rt_140.564:sq_!44_f3!65_t3!81_f3!86_t3!107_e5!113_t3!115_f3!130_t3!

Figure 8: Depiction of evidentiary reasoning from process data to observables to inferences

The task piloting methodology developed rules for assigning values to observable variables from process data. The next step was to accumulate the observable variables into a statistical model. The statistical model was required for mapping the patterns of observations from a student's performance to the student problem-solving profile which would best predict those observations. To model the student data in this study, Bayes Nets were applied. This involved specification of causal links from the profile variables to the observable variables as outlined in Table 3. For example, the *Decoding time* profile variable was modelled to predict pre-search latency observations. A technical discussion of the application of Bayes Nets is beyond the scope of this article. For a detailed account of Bayes Nets applied within technology-based educational settings, readers are referred to Korb and Nicholson (2004) and to Almond and colleagues (2007).

Communication of assessment results to relevant stakeholders is an important aspect of any assessment and reporting model (Griffin, 2007). In the problem-solving domain, it is recognised that multiple proficiencies and attributes contribute to whether and how students solve problems. After a number of observations of student performance are made, the values for the corresponding observable variables can be processed by the Bayes Net. This yields a probability distribution across the categories within each student profile variable (refer to the profile variables and categories in Table 2 for this study). For reporting purposes, this information forms a profile of the relative strengths, weaknesses and tendencies of an individual problem solver. This profile reporting approach provides ramified reports for supporting teaching and learning (Almond et al., 2007). Figure 9 shows a prototype report illustrating how the profile variables and their categories can be renamed and communicated to students, parents and educators in a user friendly format. The following paragraphs provide a brief explanation of how this report might be linked to a range of interventions which would be difficult to target without analysing process data from complex tasks.



General Problem Solving Profile

Jenny Wong

Taking time to read and analyse	[✓ Brief]	Long	OK
Understanding how to start the problems	[✓ Low]	High	
Avoiding mistakes	[✓ No]	Yes	
Searching for solutions	Limited	[✓ Lots]	Efficient
Finding solutions	Low	[✓ High]	
Time spent on problems	Brief	Long	[✓ OK]

North Shore Development Centre

NORTH SHORE Development Centre

Figure 9: Student report describing individual problem-solving characteristics

This report for the fictitious student Jenny Wong has obvious implications for instruction. Overall it provides a picture of the patterns of problem-solving behaviour that Jenny exhibited: She appeared to rush into interacting with problems; she did not start problems in a particularly goal-directed manner; she conducted some erroneous or invalid actions; her solution attempts were characterised by high redundancy; she did tend to stick at the problems until she reached the goal; and finally, she also tended to spend a reasonable amount of time engaged in searching for solutions.

Clearly there are insights here which are difficult to glean using other standalone assessment approaches.

There are a number of instructional implications arising from Jenny's report. From the first profile variable ("Taking time to read and analyse") it can be seen that she tends to spend relatively little time reading and analysing tasks prior to interacting with them. This might not be a concern if the subsequent profile variables indicated goal-directed performance, but this is not the case (refer to "Understanding how to start the problems" and "Avoiding mistakes"). Therefore the conclusion can be made that Jenny starts solving problems without investing enough time in building an understanding of the goal or the allowed object manipulations. The options for instruction at this point are numerous, and this study does not claim to define the most effective, but a few possibilities are listed: an educator could explain the importance of understanding problems before solving them; an educator could motivate Jenny by explaining that a good start will be awarded credit; an educator could explain that experts tend to spend more time at the start analysing and planning so that they can work more effectively later and get more problems right; Jenny could be given a set of problems and asked to explain or document in her own words the problem goal and rules for object manipulation prior to commencing search; Jenny could be given a set of problems and encouraged to explain how she intends to solve them and how her plan is related to the goal.

Turning now to the remaining target variables, the fact that Jenny appears to solve most problems in good time is encouraging, but the indication from the "Searching for solutions" variable is that Jenny's solutions feature some redundancy. This is consistent with her poor starts but could be targeted as a skill in its own right. The following are some suggested interventions: provide Jenny with practice tasks featuring the instruction that she should try to solve them in as few moves possible; ask Jenny to articulate why her next intended move is valid and how it gets her closer to the problem goal while she is solving problems; provide Jenny with exercises that foster the use of specific, search-based strategies such as space splitting, pairwise comparison, systematic trial and error, recursive sub-goals, etc. These suggestions are made in broad terms, and work is currently being undertaken to refine them and to add new intervention strategies suited to other student profiles.

Discussion

In this article a number of advantages of the TEM tool are mooted. These are based on assumptions that have found some support in recent decades. The first assumption is that subject matter experts find visual tools easier to work with than other forms of information (Wang, 2003). The second is that temporal information plays an important evidentiary role in several domains involving problem solving (Paek, 2002). Studies are still needed to compare the useability of the maps with existing tabulations of CTA data, if not to confirm improved interpretations due to the temporal representation, then perhaps to confirm improved useability. To evaluate the reliability of the TEM tool, a follow up study examining inter-rater reliability during code assignment would also be useful.

In developing technology-based assessment systems there are many design decisions which need to be made. The system described in this paper is in its infancy, and while an operational version exists, ongoing refinement is inevitable. This paper focuses on illustrating the potential benefits of adopting technology-rich assessments featuring complex tasks which record process data. In particular, focussing on process data is shown to enable inferences to be made about procedural aspects of performance. These inferences are attractive for their diagnostic potential and the possibility that they may provide useful, additional information from which educators might devise instructional programs. Many details concerning the probabilistic modelling of the assessment data are beyond the scope of this paper. Also, further work is presently being undertaken to formalise links from a range of profiles to appropriate interventions.

Conclusion

This paper describes an approach for analysing process data so that inferences about procedural aspects of student problem solving can be made. It is argued that *Temporal Evidence Maps* could enhance the ease of making valid interpretations of complex student-task interaction data, by re-emphasising the temporal dimension in a domain where patterns of latencies carry evidentiary weight. This could benefit assessment design teams in their efforts to formalise data evaluation rules and verify that the data relate to the intended set of student proficiencies. With considerable attention being paid to evidence observation and interpretation in modern assessment design (Pellegrino, 2002), tools such as *Temporal Evidence Maps* should boast some utility. This paper also illustrates the end product of the assessment phase in the form of a diagnostic profile report. The target inferences focussed on procedural characteristics of problem solving, providing educators with seldom reported information about student performance and cognition. Importantly, reports of this nature provide educators with another source of information for determining an individual student's instructional needs.

References

Almond, R. G., DiBello, L. V., Moulder, B. & Zapata-Rivera, J.-D. (2007). Modeling diagnostic assessments with Bayesian networks. *Journal of Educational Measurement*, 44(4), 341-359.

Bennett, R. E., Persky, H., Weiss, A. & Jenkins, F. (2007). *Problem solving in technology-rich environments: A report from the NAEP Technology-Based Assessment Project (NCES 2007–466).* Washington, DC: U.S. Department of Education. National Center for Education Statistics. http://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=2007466

Chi, M. T. H., Feltovich, P. J. & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5, 121-152.

Chi, M. T. H., Glaser, R. & Farr, M. J. (Eds.) (1988). *The nature of expertise*. Hillsdale, NJ: Erlbaum.

Chung, G. K. W. K. & Baker, E. L. (2003). An exploratory study to examine the feasibility of measuring problem-solving processes using a click-through interface. *Journal of Technology, Learning, and Assessment*, 2(2). http://escholarship.bc.edu/jtla/vol2/2/

Chung, G. K. W. K., de Vries, L. F., Cheak, A. M., Stevens, R. H. & Bewley, W. L. (2002). Cognitive process validation of an online problem solving assessment. *Computers in Human Behaviour*, 18, 669-684.

Ericsson, K. A., & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data.* Revised ed. Cambridge, MA: The MIT Press.

Flaherty, E. G. (1975). The thinking aloud technique and problem solving ability. *Journal of Educational Research*, 68(6), 223-225.

Fum, D. & Del Missier, F. (2001). Adaptive selection of problem solving strategies. In *Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society* (pp. 313-318). Mahwah, NJ: Lawrence Erlbaum Associates.

Gerace, W. J. (2001). Problem solving and conceptual understanding. [verified 23 Jun 2010] http://srri.umass.edu/files/gerace-2001psc.pdf

Gick, M. L. (1986). Problem-solving strategies. *Educational Psychologist,* 21, 99-120.

Glaser, R. (1991). Expertise and assessment. In M. C. Wittrock & E. L. Baker (Eds.), *Testing and cognition* (pp. 17-30). Englewood Cliffs, NJ: Prentice Hall.

Goldman, S., Zech, L., Biswas, G., Noser, T., Bateman, H., Bransford, J., et al. (1999). Computer technology and complex problem solving: Issues in the study of complex cognitive activity. *Instructional Science,* 27, 235-268.

Griffin, P. (2007). The comfort of competence and the uncertainty of assessment. *Studies In Educational Evaluation,* 33(1), 87-99.

Gvozdenko, E. & Chambers, D. (2007). Beyond test accuracy: Benefits of measuring response time in computerised testing. *Australasian Journal of Educational Technology*, 23, 542-558. http://www.ascilite.org.au/ajet/ajet23/gvozdenko.html

Korb, K., & Nicholson, A. E. (2004). *Bayesian artificial intelligence*. London: Chapman & Hall.

Lawson, M. J. & Chinnappan, M. (1994). Generative activity during geometry problem solving: Comparison of the performance of high-achieving and low-achieving students. *Cognition and Instruction,* 12(1), 61-93.

Leighton, J. & Gierl, M. J. (2007). Verbal reports as data for cognitive diagnostic assessment. In J. Leighton & M. J. Gierl (Eds.), *Cognitive diagnostic assessment for education* (pp. 146-172). Cambridge, UK: Cambridge University Press.

Lohman, D. F. & Ippel, M. J. (1993). Cognitive diagnosis: From statistically based assessment toward theory-based assessment. In J. R. Frederiksen, R. J. Mislevy & I. I. Bejar (Eds.), *Test theory for a new generation of tests* (pp. 41-71). Hillsdale, NJ: Lawrence Erlbaum Associates.

Luchins, A. S. (1942). Mechanization in problem solving. *Psychological Monographs,* 54.

Masters, J. (2010). Automated scoring of an interactive geometry item: A proof-of-concept. *Journal of Technology, Learning, and Assessment,* 8(7). http://escholarship.bc.edu/jtla/vol8/7/

Marshall, S. P. (1995). *Schemas in problem solving*. New York: Cambridge University Press.

Mayer, R. E. & Wittrock, M. C. (1996). Problem-solving transfer. In D. C. Berliner & R. C. Calfee (Eds.), *Handbook of educational psychology* (pp. 411-452). New York: Simon & Schuster Macmillan.

Mills, C. N., Potenza, M. T., Fremer, J. J. & Ward, W. C. (2002). *Computer-based testing: Building the foundation for future assessments*. New Jersey: Laurence Erlbaum Associates, Inc.

Mislevy, R. J. (1993). Foundations of a new test theory. In J. R. Frederiksen, R. J. Mislevy & I. I. Bejar (Eds.), *Test theory for a new generation of tests* (pp. 19-39). Hillsdale, NJ: Lawrence Erlbaum Associates.

Mislevy, R. J., Steinberg, L. S., Breyer, F. J., Almond, R. G. & Johnson, L. (1999). A cognitive task analysis with implications for designing simulation-based performance assessment. *Computers in Human Behaviour,* 15(3-4), 335-374.

Mislevy, R. J. (2008). Some implications of expertise research for educational assessment. Paper presented at the 34th International Association for Educational Assessment (IAEA) Conference. [verified 3 Jun 2010] http://www.education.umd.edu/EDMS/mislevy/papers/MislevyIAEA2008.doc

Montague, W. E. & Applegate, B. (1993). Middle school students' mathematical problem solving: An analysis of think-aloud protocols. *Learning Disability Quarterly,* 16(1), 19-32.

Newell, A. & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, N.J.: Prentice-Hall.

Paek, P. L. (2002). Problem solving strategies and metacognitive skills on SAT mathematics items. (Doctoral dissertation, University of California, Berkeley, 2002). *Dissertation Abstracts International, 63*(9), 3139.

Pellegrino, J. W. (2002). Knowing what students know. *Issues in Science & Technology,* 19(2), 48.

Pelligrino, J., Chudowsky, N. & Glaser, R. (Eds.). (2001). *Knowing what students know: The science and design of educational assessment.* Washington, DC: National Academy Press. http://www.nap.edu/catalog.php?record_id=10019

Polya, G. (1945). *How to solve it* (1st ed.). Princeton: Princeton University Press.

Richardson, M., Baird, J.-A., Ridgway, J., Ripley, M., Shorrocks-Taylor, D. & Swan, M. (2002). Challenging minds? Students' perceptions of computer-based World Class Tests of problem solving. *Computers in Human Behaviour,* 18, 633-649.

Ridgway, J. & McCusker, S. (2003). Using computers to assess new educational goals. *Assessment in Education: Principles, Policy & Practice,* 10(3), 309-328.

Robertson, I. S. (2001). *Problem solving*. Philadelphia: Psychology Press.

Schnipke, D. L. & Scrams, D. J. (1999). *Exploring Issues of test taker behaviour: Insights gained from response-time analyses.* (No. 98-09): Law School Admission Council.

Schoenfeld, A. H. (1985). *Mathematical problem solving.* New York: Academic Press.

Sternberg, R. J. & Ben-Zeev, T. (2001). *Complex cognition: The psychology of human thought*. New York: Oxford University Press.

Stevens, R. H. & Thadani, V. (2006). A Bayesian network approach for modeling the influence of contextual variables on scientific problem solving. In M. Ikeda, K. Ashley & T.-W. Chan (Eds.), *Proceedings of the 8th International Conference on Intelligent Tutoring Systems (ITS 2006).* Berlin: Springer-Verlag.

van der Linden, D., Sonnentag, S., Frese, M. & van Dyck, C. (2001). Exploration strategies, performance, and error consequences when learning a complex computer task. *Behaviour and Information Technology*, 20(3), 189-198.

Vendlinski, T. & Stevens, R. (2002). Assessing student problem-solving skills with complex computer-based tasks. *Journal of Technology, Learning, and Assessment,* 1(3). http://escholarship.bc.edu/jtla/vol1/3/

Wang, N. (2003). Use of the Rasch IRT model in standard setting: An item-mapping method. *Journal of Educational Measurement,* 40(3), 231-251.

Williamson, D. M., Bauer, M., Steinberg, L. S., Mislevy, R. J., Behrens, J. T. & DeMark, S. F. (2004). Design rationale for a complex performance assessment. *International Journal of Testing,* 4(4), 303-332.

Williamson, D. M., Bejar, I. I. & Mislevy, R. J. (2006). Automated scoring of complex tasks in computer-based testing: an introduction. In D. M. Williamson, I. I. Bejar & R. J. Mislevy (Eds.), *Automated scoring of complex tasks in computer-based testing* (pp. 1-13). New Jersey: Laurence Erlbaum Associates, Inc.

Wirth, J. & Klieme, E. (2003). Computer-based assessment of problem solving competence. *Assessment in Education,* 10(3), 329-345.

Wu, M., Griffin, P., Dulhunty, M. & Mak, A. (2002, December). Teaching strategies in problem solving. Paper presented at the AARE Conference, Brisbane. http://www.aare.edu.au/02pap/gri02632.htm

Wu, M. & Adams, R. J. (2006). Modelling mathematics problem solving item responses using a multidimensional IRT model. *Mathematics Education Research Journal,* 18(2), 93-113. http://www.merga.net.au/documents/MERJ_18_2_Wu.pdf

Nathan Zoanetti is a Research Fellow at the Assessment Research Centre in the Melbourne Graduate School of Education at the University of Melbourne. His research interests include assessment design and analysis projects, particularly those integrating technology.

Nathan Zoanetti, Assessment Research Centre
Melbourne Graduate School of Education
The University of Melbourne, Victoria 3010, Australia.
Email: n.zoanetti@unimelb.edu.au